A Strongly Normalizing Computation Rule for the Univalence Axiom in Higher-Order Propositional Logic

Robin Adams[1], Marc Bezem[1], and Thierry Coquard[2]

[1] Universitetet i Bergen, Bergen, Norway
{robin.adams,marc}@uib.no
[2] University of Gothenburg, Gothenburg, Sweden
coquand@chalmers.se

Homotopy type theory offers the promise of a formal system for the univalent foundations of mathematics. However, if we simply add the univalence axiom to type theory, then we lose the property of canonicity — that every term computes to a normal form. A computation becomes 'stuck' when it reaches the point that it needs to evaluate a proof term that is an application of the univalence axiom. We wish to find a way to compute with the univalence axiom.

As a first step towards such a system, we present here a system of higher-order propositional logic, with a universe Ω of propositions closed under implication and quantification over any simple type over Ω. We add a type $a =_A b$ for any terms a, b of type A (this type is not a proposition in Ω), and two ways to prove an equality: reflexivity, and the univalence axiom. We present reduction relations for this system, and prove the reduction confluent and strongly normalizing.

Predicative higher-order propositional logic with equality. We call the following type theory predicative higher-order propositional logic. It contains a universe Ω of propositions that contains $\bot$ and is closed under $\to$. The system also includes the higher-order types that can be built from Ω by $\to$. Its rules of deduction are

$$\frac{}{\langle\rangle \text{ valid}} \qquad \frac{\Gamma \text{ valid}}{\Gamma, x : A \text{ valid}} \qquad \frac{\Gamma \vdash \phi : \Omega}{\Gamma, p : \phi \text{ valid}} \qquad \frac{\Gamma \text{ valid}}{\Gamma \vdash x : A}\ (x : A \in \Gamma) \qquad \frac{\Gamma \text{ valid}}{\Gamma \vdash p : \phi}\ (p : \phi \in \Gamma)$$

$$\frac{\Gamma \text{ valid}}{\Gamma \vdash \bot : \Omega} \qquad \frac{\Gamma \vdash \phi : \Omega \quad \Gamma \vdash \psi : \Omega}{\Gamma \vdash \phi \to \psi : \Omega}$$

$$\frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B} \qquad \frac{\Gamma \vdash \delta : \phi \to \psi \quad \Gamma \vdash \epsilon : \phi}{\Gamma \vdash \delta\epsilon : \psi}$$

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A.M : A \to B} \qquad \frac{\Gamma, p : \phi \vdash \delta : \psi}{\Gamma \vdash \lambda p : \phi.\delta : \phi \to \psi} \qquad \frac{\Gamma \vdash \delta : \phi \quad \Gamma \vdash \psi : \Omega}{\Gamma \vdash \delta : \psi}\ (\phi \simeq \phi)$$

Extensional equality. On top of this system, we add an equality relation that satisfies univalence. We add a new judgement form, $\Gamma \vdash P : M =_A M$, to denote that P is a proof of that M and N are equal terms of type A. We also add the following constructions:

- For any $M : A$, a proof $\mathsf{ref}\,(M) : M =_A M$.
- **Univalence.** Given proofs $\delta : \phi \to \psi$ and $\epsilon : \psi \to \phi$, a proof $\mathsf{univ}_{\phi,\psi}\,(\delta, \epsilon) : \phi =_\Omega \psi$.
- Given a proof $P : \phi =_\Omega \psi$, proofs $P^+ : \phi \to \psi$ and $P^- : \psi \to \phi$.
- Given a proof $\Gamma, x : A, y : A, e : x =_A y \vdash P : Mx =_B Ny$, a proof $\lambda\!\!\lambda\!\!\lambda e : x =_A y.P : M =_{A \to B} N$. (Here, e, x and y are bound within P.)
- Rules to ensure that the equality is a congruence for $\to$ and application.

The reduction relation. We define the following reduction relation on proofs and equality proofs.

$$(\mathsf{ref}\,(\phi))^{+} \rightsquigarrow \lambda x : \phi.x \qquad (\mathsf{ref}\,(\phi))^{-} \rightsquigarrow \lambda x : \phi.x$$
$$\mathsf{univ}_{\phi,\psi}\,(\delta,\epsilon)^{+} \rightsquigarrow \delta \qquad \mathsf{univ}_{\phi,\psi}\,(\delta,\epsilon)^{-} \rightsquigarrow \epsilon$$

$$(\mathsf{ref}\,(\phi) \to \mathsf{univ}_{\psi,\chi}\,(\delta,\epsilon)) \rightsquigarrow \mathsf{univ}_{\phi\to\psi,\phi\to\chi}\,(\lambda f : \phi \to \psi.\lambda x : \phi.\delta(fx), \lambda g : \phi \to \chi.\lambda x : \phi.\epsilon(gx))$$
$$(\mathsf{univ}_{\phi,\psi}\,(\delta,\epsilon) \to \mathsf{ref}\,(\chi)) \rightsquigarrow \mathsf{univ}_{\phi\to\chi,\psi\to\chi}\,(\lambda f : \phi \to \chi.\lambda x : \psi.f(\epsilon x), \lambda g : \psi \to \chi.\lambda x : \phi.g(\delta x))$$
$$(\mathsf{univ}_{\phi,\psi}\,(\delta,\epsilon) \to \mathsf{univ}_{\phi',\psi'}\,(\delta',\epsilon')$$
$$\rightsquigarrow \mathsf{univ}_{\phi\to\phi',\psi\to\psi'}\,(\lambda f : \phi \to \phi'.\lambda x : \psi.\delta'(f(\epsilon x)), \lambda g : \psi \to \psi'.\lambda y : \phi.\epsilon'(g(\delta y)))$$

$$(\mathsf{ref}\,(\phi) \to \mathsf{ref}\,(\psi)) \rightsquigarrow \mathsf{ref}\,(\phi \to \psi) \qquad \mathsf{ref}\,(M)\,\mathsf{ref}\,(N) \rightsquigarrow \mathsf{ref}\,(MN)$$
$$(\mathsf{ref}\,(\lambda x : A.M))P \rightsquigarrow \{P/x\}M \qquad (P \text{ a normal form not of the form } \mathsf{ref}\,(\_))$$
$$(\lambda\!\!\lambda\!\!\lambda e : x =_A y.P)Q \rightsquigarrow [M/x, N/y, Q/e]P \qquad (Q : M =_A N)$$

Here, $\{P/x\}M$ is an operation called *path substitution* defined such that, if $P : N =_A N'$ then $\{P/x\}M : [N/x]M = [N'/x]M$.

We can prove the following result about the canonical forms in this system:

Proposition 1. *Every closed normal form of type $\phi =_\Omega \psi$ either has the form $\mathsf{ref}\,(\_)$ or $\mathsf{univ}(\_,\_)$. Every closed normal form of the type $M =_{A\to B} N$ either has the form $\mathsf{ref}\,(\_)$ or is a $\lambda\!\!\lambda\!\!\lambda$-term.*

Thus, once we have proved strong normalization, we know that a well-typed computation never gets 'stuck' at an application of the univalence axiom.

Proof of strong normalization. The proof of strong normalization follows the method of Tait [1] We define the set of *computable* terms $E_\Gamma(A)$ for each type A, and computable proofs $E_\Gamma(M =_A N)$ for any terms $\Gamma \vdash M, N : A$

Tait's proof relies on confluence, which does not hold for this reduction relation in general. But we do have that the following, which turns out to be sufficient.

Proposition 2. *Reduction is locally confluent. All computable terms are strongly normalizing and confluent. The computability predicates are closed under reduction and well-tiped expansion.*

Theorem 3. *If $\Gamma \vdash M : A$ then $M \in E_\Gamma(A)$. If $\Gamma \vdash P : M =_A N$ then $P \in E_\Gamma(M =_A N)$.*

It follows that this system is strongly normalizing.

In the proof, we prove confluence 'on-the-fly'. That is, whenever we require a term to be confluent, the induction hypothesis provides us with the fact that that term is computable, and hence strongly normalizing and confluent.

References

[1] W. W. Tait. Intensional iinterpretation of ffunctional of finite type i. *Journal of Symbolic Logic*, 32:198–212, 1967.