Computational Paths and the Fundamental Groupoid of a Type

Arthur F. Ramos[1], **Ruy J. G. B. de Queiroz**[2], **Anjolina G. de Oliveira**[2]

[1]Microsoft – Redmond, USA

[2]Centro de Informática – Universidade Federal de Pernambuco (UFPE)

arfreita@microsoft.com {ruy,ago}@cin.ufpe.br

Abstract. *Using computational paths as the fundamental concept, we show that we can leverage Category Theory to propose the concept of fundamental groupoid of a type.*

1. Introduction

When the topic is type theory, the two most widely known theories are Alonzo Church's typed λ-calculus and Per Martin-Löf's intuitionistic type theory. Here we are particularly interested in the latter one. Intuitionistic type theory has two main variations, the intensional and extensional ones. The intensional version comes with a formulation of identity type which allows for the incorporation of paths in the syntax, and this helps to make a bridge connecting Type Theory with Homotopy Theory. This connection was responsible for the formulation of a new theory called Homotopy type theory [Awodey 2012]. Since its formulation, it has rapidly become an area of intensive research. The main reason for that was the discovery of Univalent Models by Vladimir Voevodsky in 2005 [Voevodsky 2014]. This discovery showed that homotopy type theory is not just a computational type theory, but also an important theory of mathematics. In fact, Voevodsky himself argued that this theory is more suitable for working as the foundations of mathematics than the well-established set theory [Voevodsky 2014].

The identity type is the bridge that connects type theory and homotopy theory. In type theory, given a type A and elements a and b of type A, we can think of an identity type $Id_A(a,b)$ as the type of proofs that a and b are equal. That way, every element p (also known as the witness p) of the type $Id_A(a,b)$ can be interpreted as a proof that establishes that a and b are propositionally equal [Univalent Foundations Program 2013]. Within the homotopy interpretation, the type A is considered as a topological space A and the elements a and b are considered points of A. Furthermore, the element p of $Id_A(a,b)$ is a homotopical path between points a and b [Univalent Foundations Program 2013]. Adding this interpretation to the intuitionistic type theory forms the homotopy type theory. Homotopy type theory has some groundbreaking results, as one can check in [Univalent Foundations Program 2013]. In this work, the main interest in homotopy type theory is its relation with an entity called computational paths. This relation, together with a formal definition of a computational path, are given in the following subsection.

2. Computational Paths

A computational path, also known as a sequence of rewrites, is an entity that arises from the notion that two computational objects are equal iff they have a computational path connecting them. Before we try to formally define a computational path, we need to recall some concepts of λ-*calculus*. In the theory of λ-*calculus*, we have a formal theory called $\lambda\beta\eta$ theory of $\lambda\beta\eta$-equality [Hindley and Seldin 2008]. This theory consists of axioms and rules of inferences. Using these axioms and rules, we can formally show the equality of two λ-formulas, $M = N$ [Hindley and Seldin 2008]. We are interested in an equality theory for Martin-Löf's type theory though. Based in the theory of $\lambda\beta\eta$-equality, we can obtain the following axioms for Martin-Löf's type theory [Ramos et al. 2017]:

$$(\beta)\quad \dfrac{N : A \qquad \begin{array}{c}[x : A]\\ M : B\end{array}}{(\lambda x.M)N = M[N/x] : B} \qquad\qquad (\xi)\quad \dfrac{\begin{array}{c}[x : A]\\ M = M' : B\end{array}}{\lambda x.M = \lambda x.M' : (\Pi x : A)B}$$

$$(\rho)\quad \dfrac{M : A}{M = M : A} \qquad\qquad (\mu)\quad \dfrac{M = M' : A \qquad N : (\Pi x : A)B}{NM = NM' : B}$$

$$(\sigma)\quad \dfrac{M = N : A}{N = M : A} \qquad\qquad (\upsilon)\quad \dfrac{N : A \qquad M = M' : (\Pi x : A)B}{MN = M'N : B}$$

$$(\tau)\quad \dfrac{M = N : A \qquad N = P : A}{M = P : A}$$

$$(\eta)\quad \dfrac{M : (\Pi x : A)B}{(\lambda x.Mx) = M : (\Pi x : A)B}\ (x \notin FV(M))$$

Aside from these axioms, another important operation is the change of bound variables, denoted by α-equality [Ramos et al. 2017].

Definition 2.1 (Computational path [Ramos et al. 2017]) Let a and b be elements of a type A. A *Computational path* from a to b is a sequence of definitional equalities (each definitional equality is either an application of an equality axiom or a change of bound variables) starting in a and arriving at b.

As an example, take the reduction $(\lambda y.yv)z : A \rhd_{\beta} zv : A$. We say that $(\lambda y.yv)z : A$ is equal to $zv : A$ because there is a computational path $\beta((\lambda y.yv)z, zv)$ that establishes this equality. Moreover, the equality axiom τ can be used to compose paths. So, for example, take the reductions $(\lambda x.(\lambda y.yx)z)v \rhd_{\beta} (\lambda y.yv)z \rhd_{\beta} zv$. This time we had to apply β axiom two times. The computational path that establishes the equality between $(\lambda x.(\lambda y.yx)z)v$ and zv will then be the composition of the two β paths. This composition can be made by applying the τ axiom, resulting in the path $\tau(\beta, \beta)$, which establishes the equality between $(\lambda x.(\lambda y.yx)z)v$ and zv.

2.1. Rewrite system and *rw*-equalities

One essential aspect of computational paths is that they carry a notion of reduction and normal form. A reduction happens when we remove a redundancy from a computational path. To see this, consider a path p. Starting from p, we can apply the rewrite σ(symmetry) twice in a row, obtaining $\sigma(\sigma(p))$. We are inverting p twice. That way, starting at p and inverting it two times will result in ending at p. Therefore, we consider that $\sigma(\sigma(p))$ has a redundancy and that it can be reduced to p. We use the notation $\sigma(\sigma(p)) \rhd_{\mathrm{rw}} p$ to indicate that $\sigma(\sigma(p))$ reduces to p (In this case, we could use $\rhd_{\mathrm{1rw}}$ indicating that the reduction made was only one single reduction). To explain the concept of computational path reductions, we used a simple example involving symmetry. Nevertheless, there are many redundancies generated by the equality axioms and some are rather involved. A reduction rule (i.e., a rule that removes a redundancy) is known as *rw*-rule [Ramos et al. 2017].

The system with all *rw*-rules is called *LND_{EQS}-RWS* [de Queiroz et al. 2016]. This system is also terminating and confluent. The formal proofs of these properties are given in [de Queiroz and de Oliveira 1994, de Oliveira and de Queiroz 1999]. The subset of rules used in this work are:

Rules involving σ and ρ: $\sigma(\rho) \rhd_{sr} \rho \quad \sigma(\sigma(r)) \rhd_{ss} r$
Rules involving τ: $\tau(r,\sigma(r)) \rhd_{tr} \rho \quad \tau(\sigma(r),r) \rhd_{tsr} \rho \quad \tau(r,\rho) \rhd_{trr} r \quad \tau(\rho,r) \rhd_{tlr} r$
Rule involving τ and τ: $\tau(\tau(t,r),s) \rhd_{tt} \tau(t, \tau(r,s))$

Definition 2.2 (*rw*-equality [Ramos et al. 2017]) Let a and b be computational paths. We say that $a =_{rw} b$ (read as: a is rw-equal to b) iff b can be obtained from a by a finite (perhaps empty) series of *rw*-contractions and reversed *rw*-contractions. In other words, $a =_{rw} b$ iff there exists a sequence $R_0,....,R_n$, with $n \geq 0$, such that $(\forall i \leq n - 1)$ $(R_i \rhd_{1rw} R_{i+1}$ *or* $R_{i+1} \rhd_{1rw} R_i)$, where $R_0=a$ and $R_n=b$.

3. Fundamental Groupoid of a type

In a previous work [Ramos et al. 2017] we have noticed that a computational path follows a structure which is very similar to the one of a category. Informally speaking, a category is a mathematical entity composed by objects and morphisms (arrows) between the objects. These morphisms can be composed under certain conditions and this composition must be associative, i.e., $(f \circ g) \circ h = f \circ (g \circ h)$, and for every object there exists an identity morphism. One of the most interesting parts is that the structure that we have achieved is not, in the strict sense, a category. The reason for that is the fact that the composition properties do not hold under equality. Nonetheless, we have concluded that the properties held when we changed the equality for *rw*-equality. In this sense, the properties hold under *rw*-equality. In the literature, we consider such structure as being a **weak** category.

Definition 3.1 (A_{rw} [Ramos et al. 2017]) A_{rw} is a structure composed by objects and morphisms. The objects are elements $a : A$ and the morphisms (also known as arrows) are computational paths between two objects.

A groupoid is a category for which every arrow is an isomorphism [Awodey 2010]. An arrow s: $a \rightarrow b$ is considered an isomorphism if it has an arrow t: $b \rightarrow a$ such that $\tau(s,t) = \rho_a$ and $\tau(t,s) = \rho_b$. Since these equalities will hold only up to *rw*-equality, we propose the following:

Proposition 3.1 ([Ramos et al. 2017]) A_{rw} has a weak groupoid structure.

Proof First, we need to prove that A_{rw} is a weak category. To do that, we need to define composition of morphisms and the identity arrow. For that, let composition of paths $s \circ t$ be given by application of the transitivity identifier, i.e., $s \circ t = \tau(t, s)$. The identity morphism of an object a is given by the reflexive path $a =_{\rho_a} a$. We need now to check the associative and identity laws. The associativity equation holds weakly, we just need to use the *tt* rule: $\tau(\tau(s,r),t) =_{rw_{tt}} \tau(s, \tau(r,t))$.

Using rules tlr and trr, we show that the identity laws also hold weakly: $s \circ 1_a = s \circ \rho_a = \tau(\rho_a, s) =_{rw_{tlr}} s$ and $1_b \circ s = \rho_b \circ s = \tau(s, \rho_b) =_{rw_{trr}} s$.

With these conditions satisfied, we conclude that A_{rw} is indeed a weak category. Now we need to show that it is a weak groupoid. For that, we need to show that every computational path s has an inverse computational path s'. Finding s' is easy, just put $s' = \sigma(s)$.

To show that the equalities of the isomorphism hold weakly, we use rules tr and tsr: $s \circ s' = s \circ \sigma(s) = \tau(\sigma(s), s) =_{rw_{tsr}} \rho_b$ and $s' \circ s = \sigma(s) \circ s = \tau(s, \sigma(s)) =_{rw_{tr}} \rho_a$. $\square$

With that, we conclude that every type A has a underlying groupoid structure A_{rw}. We call this structure the *fundamental groupoid of the type A*. In a sense, using the topological interpretation of a type, one can think of the type A as the equivalent of a space A and the fundamental groupoid of type A as the fundamental group of the space A. This concept of a fundamental groupoid gives rise to interesting results. First, one can use the concept of fundamental groupoid to show, using an entire computational interpretation, that it is possible to prove that the fundamental groupoid of the circle S^1 is equivalent to the integers $\mathbb{Z}$ [Ramos et al. 2018]. Another possible conclusion is that one can think of computational paths of computational paths, which generates categories of higher dimensions. That makes possible interpretation of groupoids of higher structures, as showed in [Ramos et al. 2017]. With that in mind, we conclude that the fundamental groupoid of a type is a **key concept** that gives rise to multiple lines of further research.

References

Awodey, S. (2010). *Category theory*. Oxford University Press.

Awodey, S. (2012). Type theory and homotopy. In Dybjer, P., Lindström, S., Palmgren, E., and Sundholm, G., editors, *Epistemology versus Ontology*, volume 27 of *Logic, Epistemology, and the Unity of Science*, pages 183–201. Springer Netherlands.

de Oliveira, A. G. and de Queiroz, R. J. G. B. (1999). A normalization procedure for the equational fragment of labelled natural deduction. *Logic Journal of IGPL*, 7(2):173–215.

de Queiroz, R. J. G. B. and de Oliveira, A. G. (1994). Term rewriting systems with labelled deductive systems. In *Proceedings of Brazilian Symposium on Artificial Intelligence (SBIA'94)*, pages 59–72.

de Queiroz, R. J. G. B., de Oliveira, A. G., and Ramos, A. F. (2016). Propositional equality, identity types, and direct computational paths. *South American Journal of Logic*, 2(2):245–296. Special Issue A Festschrift for Francisco Miraglia, M. E. Coniglio, H. L. Mariano and V. C. Lopes (Guest Editors).

Hindley, J. R. and Seldin, J. P. (2008). *Lambda-calculus and combinators: an introduction*. Cambridge University Press.

Ramos, A. F., De Queiroz, R. J. G. B., and De Oliveira, A. G. (2017). On the identity type as the type of computational paths. *Logic Journal of the IGPL*, 25(4):562–584.

Ramos, A. F., de Queiroz, R. J. G. B., de Oliveira, A. G., and De Veras, T. M. L. (2018). Explicit computational paths. *South American Journal of Logic*, 4(2):441–484. Proceedings of the XVIII Brazilian Logic Conference.

Univalent Foundations Program, T. (2013). *Homotopy Type Theory: Univalent Foundations of Mathematics*. https://homotopytypetheory.org/book, Institute for Advanced Study.

Voevodsky, V. (2014). Univalent foundations and set theory. Univalent Foundations and Set Theory, Lecture at IAS, Princeton, New Jersey, Mar 2014.