

# A FORMAL CALCULUS OF PROFUNCTORS IN HOMOTOPY TYPE THEORY USING AGDA

## A PART III PROJECT PROPOSAL

Rupert Horlick (rh572)  
Homerton College

Prof M. Fiore (mpf23)  
Christ's College

This project aims to formalise the theory of generalised species of structures [2] in a proof assistant. To do this we must first formalise the calculus of profunctors and the free symmetric-monoidal completion construction. All of this will be done in the context of Homotopy Type Theory [8], which provides a novel and natural foundation for formalising mathematical structures. The project will be written in Agda [7], on top of the HoTT-Agda [3] library.

## 1 INTRODUCTION, APPROACH AND OUTCOMES

The main aim of this project is to formalise the theory of *generalised species of structures* [2].

A *combinatorial species* [4] can be thought of intuitively as the set of shapes of a given datatype. For example, the set of all rooted binary trees with a finite set of nodes. Formally this is a functor from  $\mathbf{B}$ , the groupoid of finite sets and bijections, to  $\mathbf{Set}$ , the category of sets and functions. A *generalised species* replaces  $\mathbf{B}$  with  $!\mathbb{A}$ , for some small category  $\mathbb{A}$ , and  $\mathbf{Set}$  with  $\widehat{\mathbb{B}} = \mathbf{Set}^{\mathbb{B}^{\text{op}}}$ , for some small category  $\mathbb{B}$ . The  $!$  construction on  $\mathbb{A}$  is the *free symmetric-monoidal completion*. This is analogous to the free commutative-monoid construction on sets, given by the finite-multiset construction. A functor  $\mathbb{A} \rightarrow \widehat{\mathbb{B}}$  is equivalent to a profunctor [1]  $\mathbb{A} \nrightarrow \mathbb{B}$ , that is a functor  $\mathbb{B}^{\text{op}} \times \mathbb{A} \rightarrow \mathbf{Set}$ . Before implementing the calculus of generalised species we must first implement the *calculus of profunctors* and the  $!$  construction.

The calculus of profunctors provides operations on profunctors together with algebraic laws that hold between them. We are interested in operations such as composition, product, and sum of profunctors. Furthermore, when the domain of the profunctor is a free symmetric-monoidal category

Categorical Notion		Type
Groupoid	$ \mathbb{G} $	$A$
	$\text{Hom}_{\mathbb{G}}$	$\text{Id}_A$
Presheaf	$\widehat{\mathbb{G}}$	$A \rightarrow \mathcal{U}$
Profunctor	$\mathbb{G} \rightharpoonup \mathbb{H}$	$A \rightarrow B \rightarrow \mathcal{U}$
End	$\int_c$	$\Pi_c$
Coend	$\int^c$	$\Sigma_c$

Table 1: Correspondence between categorical notions and types

we can also define differentiation. We can then prove laws such as Leibniz's rule and the chain rule. To define the various operations, we will first need to formalise categorical notions such as Kan extensions and (co)ends.

For a small category  $\mathbb{C}$ ,  $!\mathbb{C}$  is its free symmetric-monoidal completion. This can be explicitly defined as the category whose objects are finite sequences  $\langle c_i \rangle$  of objects of  $\mathbb{C}$  and whose morphisms are pairs of a permutation  $\sigma$  and a sequence of maps  $\langle f_i : c_i \rightarrow c'_{\sigma i} \rangle$ . It is not currently clear how this will be formalised and, in fact, this represents the first significant challenge of the project.

*Homotopy Type Theory* (HoTT) [8] is a new foundation for mathematics, based on Martin-Löf Type Theory [6]. It interprets types as spaces in the topological sense and inhabitants of a type as points in the space. Two points in the space are equal if there is a *path* between them. The paths between two points  $a$  and  $a'$  of  $A$  constitute the identity type  $\text{Id}_A(a, a')$ . We can then have paths between paths and so on. Each type therefore has an  $\infty$ -*groupoid* structure. This makes HoTT a natural place to embed groupoidal notions. We will do just that, implementing the calculus of generalised species (for groupoids) internally, rather than implementing it in a Category Theory library on top of HoTT. This relies on the correspondence between categorical notions and type-theoretic ones shown in Table 1 (cf. [5]).

The core of the project is the formalisation of the calculus of profunctors. Once that has been completed, we will attempt to find a construction corresponding to the free symmetric-monoidal completion. If this is successful, then we will formalise the theory of generalised species internally in the type theory. If we cannot formalise the  $!$  construction internally, then we will try to do it on top of a Category Theory library. We can then implement the theory of generalised species at that level instead.

## 2 WORKPLAN

- 28/11/2016** Background reading. I will need to learn enough Category Theory and Homotopy Type Theory to be able to implement the constructions described above. I will also need to learn to use Agda, which I have not used before.
- 05/12/2016** Continue reading. It is not likely that I will be able to read all the necessary material, even in the first four weeks, but I will continue to read alongside development after this point.
- 19/12/2016** Internal implementation of background [enriched] categorical notions (I): presheaves, Yoneda, cartesian closed structure of presheaf categories, (co)ends, [weighted] (co)limits, etc.
- 02/01/2017** Internal implementation of background [enriched] categorical notions (II): Kan extensions, geometric realization, promonoidal structure and Day convolution, etc.
- 16/01/2017** Internal implementation of the calculus of profunctors. The main property we want to prove is that profunctors form a compact closed bicategory with biproducts.
- 30/01/2017** Explore how to implement the free symmetric monoidal category construction. If we cannot figure this out then we will take the project in a different direction.
- 13/02/2017** Finish implementing the free symmetric monoidal category construction and prove all the needed combinatorial facts about it.
- 27/02/2017** Begin implementation of the theory of generalised species of structures.
- 13/03/2017** Finish implementation of generalised species.
- 27/03/2017** Prove that generalised species form a cartesian closed bicategory.
- 10/04/2017** Explore the theories of operads and opetopes in the cartesian closed bicategory of generalised species of structures.
- 24/04/2017** Write up
- 08/05/2017** Write up and try to finish first draft
- 22/05/2017** Redraft and hand in a week ahead of time!

## REFERENCES

- [1] Jean Bénabou. Les distributeurs, 1973.
- [2] Marcelo Fiore, Nicola Gambino, Martin Hyland, and Glynn Winskel. The cartesian closed bi-category of generalised species of structures. *Journal of the London Mathematical Society*, 77(1): 203–220, 2008.
- [3] Kuen-Bang Hou (Favonia). Homotopy Type Theory in Agda. <https://github.com/HoTT/HoTT-Agda>, 2016.
- [4] André Joyal. Foncteurs analytiques et especes de structures. In *Combinatoire énumérative*, pages 126–159. Springer, 1986.
- [5] F William Lawvere. Metric spaces, generalized logic, and closed categories. *Rendiconti del seminario matematico e fisico di Milano*, 43(1):135–166, 1973.
- [6] Per Martin-Löf. An intuitionistic theory of types: Predicative part. *Studies in Logic and the Foundations of Mathematics*, 80:73–118, 1975.
- [7] Ulf Norell. *Towards a practical programming language based on dependent type theory*, volume 32. Citeseer, 2007.
- [8] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. <https://homotopytypetheory.org/book>, Institute for Advanced Study, 2013.