Formalizing Coq Modules in the MetaCoq project

Bachelor's Thesis

Yee-Jian Tan

October 5, 2022

The harmony of the world is made manifest in Form and Number, and the heart and soul and all the poetry of Natural Philosophy are embodied in the concept of mathematical beauty.

– D'Arcy Wentworth Thompson

Abstract

This is an abstract. I am formalizing Coq modules in MetaCoq, it is a missing piece of the already established MetaCoq project. In the second part, as a requirement for the mathematics requirement, I study some basic properties of lambda calculus, hopefully culminating in the decidability for conversion in real-world type systems such as the Martin-Lof Type Theory (MLTT).

Acknowledgements

I wouldn't come this far with the help of many people along the way. I would like to thank especially Nicolas Tabareau for having me in Nantes for the summer of 2022, which changed my life. Thank you professors Yang Yue and Martin Henz of the National University of Singapore for being my advisors for this project, even though it a rare one — externally proposed, cross-faculty, joint final year project.

I would also like to thank the people who inadvertently made this possible — Thomas Tan Chee Kun who gave me a foray into the field of Type Theory, then Rodolphe Lepigre who was an amazing mentor landing me into a summer of internship. Not forgetting the valuable friendship and advice I received in Gallinette team, for the people whom I interacted with: Pierre, Tomas, Yann, Arthur, Iwan, Kenji, Meven, Assia, Pierre-Marie, Hamza, Enzo, Yannick and everyone else that appeared during my stay in Gallinette. I truly loved the time there and can't wait to be back again.

Reading this paper

This paper is written with screen readers in mind. I will add links and references wherever possible, especially in any upcoming definitions. Click on symbols to jump to its definition, if I figure out how to make it work.

Contents

Part I: Formalization of Coq Modules

Introduction to the MetaCoq Project

1

MetaCoq is a project to formalize the core calculus, PCUIC, in Coq, and become a platform to write tools that can manipulate Coq terms. The effort was complete for a large part of the core language of Coq, with a few missing pieces:

- Eta
- Template Polymorphism
- SProps
- Modules

I will be tackling the last.

2 The Module System in Coq

Coq modules are not first class objects of the language. In fact, it is closer to a dictionary that stores each entries of a path into a canonical name, which is then refered to, or compared with when needed.

How do functors work? They are stored as-is in the global environment. Functor applications are evaluated applicatively/generatively.

Part II: Decidability of Conversion in MLTT

The Untyped Lambda Calculus 3

Here, I present some properties of the untyped lambda calculus

3.1 The Church-Rosser Confluence

We study the paper by Takahashi's successors on a proof for Church-Rosser with a strong property: the target of confluence depends solely on the top of the diamond, independent of the other vertices and edges of the diamond.