Working Principles Of Proof Assistants And Formalization Of Some Proofs in Agda

Suprevisior : K.B manandar

Ashwot Acharya, Bishesh Bohora, Supreme chaudary

May 30, 2025

1 Introduction

In 1998 Thomas C. Hales announced that he had proved the kepler conjecture in 1998 (Solane,1998) However the peer review took over 4 years especially due to the proof being incredibely difficult to check , with over a dozen of mathematician to refree the proof and although they were 99% certain it was not untill the proof became formalized that they were able to completely validate the correctness of the proof, which according to Thomas Hales would have taken 20 person-year of manual work.(Szpiro, G, 2003) Proof assistants helps with formalization of mathematical proofs in computer which enables for faster proofs computations.
Some proofs could have many many special cases which isn't easy to figureout using tradational methods, but with the use of proof assistant these cases can easily be accounted for. Proof assistant aren't just about guiding with complicated proofs but also can help to check if the proof that is written is correct especially for those long and tedious proofs.

2 Problem statement

This projects aims to design and implementat Haskell-Based proof assistant software and formalize some of the proofs of linear Algebra. Leveraging Haskell's strong type system and its purely functional nature.

3 Research objectives

1. To Investigate the current exsisting proof assistant software and identify the mathematics behind them.
2. To implementat Intuitionistic type theory and apply the logic of constructivism.
3. To formalize some selected proofs of Linear Algebra

4 Literature Review

Proof assistant rely extremely heavily on the type of logic it is coded on, for example

5 Methodology

Investigating the Coq and the agda proof assistant and assessing the Logic that each proof assistant use and how the proof assistant use such logic to help check if the proof is valid. We will also be inverstigating how to formalize some proofs in one of the proof assistant, and seeing how the logic holds in formalization of proofs.

6 Expected Outcomes

Learning indepth about type theory, curry-howard correospondance and formalization of some proofs.

7 Significiance

Being able to Formalze proofs in compute

8 Work Plan

Week	Work plan
1	understanding the various simple type theory models
2	understanding the implementation of type theory models in digital proof assistant
3	Understanding Purely functional Programming paradigm and λ-calculus
4	Working Principles of agda and its core implementation
5	Implementation of some proofs in agda

9 Refrences

1. Sloane, N.J.A. (1998). Kepler's conjecture confirmed. Nature, 395(6701), pp.435-436 doi:https://doi.org/10.1038/26609.

2. Szpiro, G. (2003). Does the proof stack up? Nature, 424(6944), pp.12–13, doi:https://doi.org/10.1038/424012a.