

# Analysis 2 Term Paper Proposal

Yuxuan Sun

March 18, 2022

## 1 Introduction

This paper aims to explore the formalization of real analysis and possibly functional analysis in homotopy type theory using proof assistants. As modern mathematics becomes more abstract and the correctness of proofs relies on mutual-trust, verifications by computers could be helpful. Due to the characteristics of type theory, which will be explained later on, current popular proof assistants are based on it.

The ideal outline of the paper would be sufficient introductions on type theory at the beginning such that some formalizations of theorems in analysis could be understood.

## 2 type theory vs. set theory(informal)

Both homotopy type theory and Zermelo-Fraenkel set theory are foundational language for mathematics.

Set theory has two "layers": the deductive system of first-order logic, and, the axioms of a particular theory, such as ZFC. Thus, set theory is more about the interplay between sets (the objects of the second layer) and propositions (the objects of the first layer).

Type theory only has one: types. Propositions are identified with particular types.

A deductive system is a collection of rules for deriving judgments.

In the deduction system of first-order logic, there is only one kind of judgment: a given proposition has a proof.

In type theory, to say  $A$  has a proof, we will write  $a : A$ , pronounced as: the term  $a$  has type  $A$ . When  $A$  is a type representing a proposition, then  $a$  maybe called a witness to the provability of  $A$  or evidence of the truth of  $A$ .

A crucial difference is that  $a : A$  is a judgment whereas  $a \in A$  is a proposition. One could use rules to derive from a judgement yet can neither prove nor disprove a judgement([Program, 2013](#)).

### 3 Martin-Löf's Dependent Type Theory

All from [Rijke, 2021](#).

**Definition 1.** There are four kinds of judgments in Martin-Löf's dependent type theory:

1.  $A$  is a (well-formed) type in context  $\Gamma$ . We express this judgment as

$$\Gamma \vdash A \text{ type}$$

2.  $A$  and  $B$  are judgmentally equal types in context  $\Gamma$ . We express this judgment as

$$\Gamma \vdash A \doteq B \text{ type.}$$

3.  $a$  is a (well-formed) term of type  $A$  in context  $\Gamma$ . We express this judgment as

$$\Gamma \vdash a : A$$

4.  $a$  and  $b$  are judgmentally equal terms of type  $A$  in context  $\Gamma$ . We express this judgment as

$$\Gamma \vdash a \doteq b : A.$$

**Definition 2** (context). A *context* is a finite list of variable declarations

$$x_1 : A_1, x_2 : A_2(x_1), \dots, x_n : A_n(x_1, \dots, x_{n-1})$$

satisfying the condition that for each  $1 \leq k \leq n$  we can derive the judgment

$$x_1 : A_1, \dots, x_{k-1} : A_{k-1}(x_1, \dots, x_{k-2}) \vdash A_k(x_1, \dots, x_{k-1}) \text{ type}$$

using the inference rules of type theory.

Some definitions that will be included: family, inference rules, fiber. (maybe will be in appendix?)

### 4 implementation

```
data N : Set where
  zero : N
  suc  : (n : N) -> N
  _+_  : N -> N -> N
  zero + n = n
  suc m + n = suc (m + n)
```

For example, number 2 will be written as  $\text{suc}(\text{suc}(\text{zero}))$  ([Lundfall](#)).

There will also be a brief summary on the construction of rational number with an appendix of code (it's too long), referencing from [Lundfall](#) as well.

A reference for formalizing functional analysis structures is [Affeldt et al.](#).

## References

- R. Affeldt, C. Cohen, M. Kerjean, A. Mahboubi, D. Rouhling, and K. Sakaguchi. Formalizing functional analysis structures in dependent type theory. page 18.
- M. Lundfall. Formalizing Real Numbers in Agda. page 62.
- T. U. F. Program. Homotopy Type Theory: Univalent Foundations of Mathematics. *arXiv:1308.0729 [cs, math]*, Aug. 2013. URL <http://arxiv.org/abs/1308.0729>. arXiv: 1308.0729.
- E. Rijke. Introduction to Homotopy Type Theory. page 478, 2021.