Formalizing Lindenbaum-Tarski algebra for propositional logic in Cubical Agda

Caroline Roos

Supervisor: Anders Mörtberg

Degree project

Department of Mathematics
Stockholm University

June 13, 2023

Formalizing mathematics

Why formalize?

Benefits of formalizing mathematics:

- Ensure correctness
- Detect errors in tradional mathematical proofs

There are many proof assistants out there
Coq, **Agda**, Lean, Idris, ...

Propositions as types

The propositions-as-types interpretation is the direct relationship between computer programs and mathematical proofs.

Prop	Type
$\top$	unit
$\bot$	void
$\phi_1 \wedge \phi_2$	$\tau_1 \times \tau_2$
$\phi_1 \supset \phi_2$	$\tau_1 \to \tau_2$
$\phi_1 \vee \phi_2$	$\tau_1 + \tau_2$

Becuase of strong typing and dependent types, Agda makes a good proof assistant.

Agda proof assistant

Defining a datatype in Agda

```
data Bool : Type where
  true : Bool
  false : Bool
```

Bool is the name of the datatype, and **true** and **false** are its constructors.

Agda proof assistant

Functions over datatypes can be defined using pattern matching.

```
not : Bool → Bool
not true = false
not false = true
```

The type of **not** is defined as a function from **Bool** to **Bool**. The function is then defined by pattern matching on the arguments.

Agda proof assistant

A **dependent type** is a type that depends on elements of another type.
For example, the polymorphic identity function:

$$\mathsf{id} : (A : \mathsf{Type}) \to A \to A$$
$$\mathsf{id}\ A\ x = x$$

In Agda it is possible to use implicit arguments.

$$\mathsf{id'} : \{A : \mathsf{Type}\} \to A \to A$$
$$\mathsf{id'}\ x = x$$

Agda will try to infer the type for us.

Cubical Agda

Cubical Agda is an extension of Agda that incorporates features of cubical type theory.

It has native support for set quotients!

Note: We are using the agda/cubical library, which is the standard library for Cubical Agda.

Formalizing propositional logic

#TODO

- Formulas and context?
- Deduction rules (discuss choices)

Biprovability relation

Definition

$\phi \sim \psi$ if and only if $\Gamma, \phi \vdash \psi$ and $\Gamma, \psi \vdash \phi$.

The relation is defined as a pair, so we define it as a product in Agda:

```
_∼_ : Formula → Formula → Type
ϕ ∼ ψ = Γ :: ϕ ⊢ ψ × Γ :: ψ ⊢ ϕ
```

This is an equivalence relation!

Formalizing Lindenbaum-Tarski algebra

Definition

The Lindenbaum-Tarski algebra is the quotient algebra obtained by factoring the algebra of formulas by the equivalence relation $\sim$.

Define Lindenbaum-Tarski algebra in Cubical Agda using the existing definition of set quotients.

```
LindenbaumTarski : Type
LindenbaumTarski = Formula / _∼_
```

Formalizing Lindenbaum-Tarski algebra

#TODO

- Operations

The Lindenbaum-Tarski algebra is a Boolean algebra

#TODO

- A Boolean algebra is a complemented distributive lattice
- Partial order on LT
- LT is complemented distributive lattice
- Formalizing this in Cubical Agda (+superweakening)

Applications

#TODO

- Soundness?
- Usefulness of doing algebra on logic?