Rewriting under binders, comfortably

Yves Bertot

March 2025

Plan

- difficulty proving

$$\sum_{i=0}^{n} i = \sum_{i=0}^{n} \sqrt{i^2}$$

- Formal proofs have several steps,
- Math teacher proofs are very different,
- A proposed solution.

The context

- Mathematical constructions like integrals and iterated sums have *bound variables*
- From the formal point of view, a bound variable does not really exist
- Type theory promotes *Leibniz* equality as the main tool to reason modulo equality
  - especially for rewriting
- But Leibniz equality requires objects that really exist

Discrepancy in idioms

$$\sum_{i=0}^{n} i = \sum_{i=0}^{n} \sqrt{i^2}$$

- The math teacher's proof (I believe)
  - *Replace* $\sqrt{i^2}$ *with* i *in the right-hand side sum.*
  - *Note that the sum ranges over positive values*
- The formally verified proof
  1. Establish $\forall i, 0 \leq i \leq n \Rightarrow i = \sqrt{i^2}$
  2. For this, fix i such that $0 \leq i \leq n$,
  3. Then $i = \sqrt{i^2}$ (by some proof),
  4. then apply the extensionality lemma for sums:

$$\forall f g, (\forall i, 0 \leq i \leq n \Rightarrow f(i) = g(i)) \Rightarrow \sum_{i=0}^{n} f(i) = \sum_{i=0}^{n} g(i)$$

The curse of α-conversion

- There is no doubt that, if i exists and is larger than 0, $i = \sqrt{i^2}$,
- Leibniz says: if $n = m$, you can replace n with m in any formula
  - But the numbers i and $\sqrt{i}$ do not even exist in the formula

$$\sum_{i=0}^{n} \sqrt{i^2}$$

  - Bound variable names do not count for logical reasoning

$$\sum_{i=0}^{n} \sqrt{i^2} = \sum_{j=0}^{n} \sqrt{j^2}$$

  - $\sqrt{i^2}$ does not occur in the right-hand side formula!
  - So you cannot use Leibniz' principle directly

A preliminary solution

- Make the sentence *Replace* $\sqrt{i^2}$ *with* i *in the sum.* understandable by the proof system
- Do not work modulo α-conversion

1. Recognize that $\sqrt{i^2}$ is not well-formed because we are missing a variable with the name i
2. By scanning the formula, detect that i is bound in at least one place,
3. Search for instances of $\sqrt{i^2}$ in the multiple places where this may occur
4. Do this again if there are nested binding patterns
5. Every time one enters inside an operator with bound variables, apply a suitable "extensionality" theorem

Example using the solution

DEMO

dépot git, fichier d'expérience

A prototype implementation

- Required an extension of the `Elpi` meta-programming language
- Authorize passing "open terms" as argument to tactics
  - An open term is well typed in an extension of the context
  - Example, if i does not exist in the context
    $\sqrt{i^2}$ is not well-typed,
    but $\lambda i : \mathbb{R}, \sqrt{i^2}$ is well typed
- The tactic receives two open terms, which can be viewed has a *rewrite rule*
- the context is search for a subcontext where:
  - All "open variables" are accounded
  - The left-hand side of the rewrite rule occurs

Building a proof

- Default extensionality: two functions that are equal everywhere can be substituted for each other
  - Axiom `functional_extensionality` provided by Rocq
- Ad hoc extensionality: compare functions only on a subset
  - For integrals: the subset is the interval between the bounds
  - For discrete sums with integer bounds: the subset is the intersection of the integers and the interval between the bounds

Future work

- Provide a comfortable interface to add new ad-hoc extensionality principles
- Rely on `setoid rewrite`, advanced location selection
- Handle goals that are not equalities
- perform replacement modulo orders
- Find links with *observational type theory*