Formalising Lagrange's Theorem in Coq

Dan Synek

Radboud University Nijmegen

WHY?

Informal Proof

Let G be a finite group. The order of a subgroup H of G divides the order of G.

Informal Proof: Form the cosets Hg of G where $Hg := \{hg | h \in H\}$

1. The cosets form a partition, e.g. $Hg_1 = Hg_2$ or $Hg_1 \cap Hg_2 = \emptyset$: Assume $Hg_1 \cap Hg_2 \neq \emptyset$ so $h_1 g_1 = h_2 g_2$ and thus $g_1 = h_1^{-1} h_2 g_2$, then $g_1 \in Hg_2$ and thus $Hg_1 \subseteq Hg_2$ By symmetry $Hg_2 = Hg_1$.
2. Each coset has exactly $|H|$ elements: If $h_1 g = h_2 g$ then $h_1 = h_2$.

Informal Version of Formal Proof

1. $G \simeq G/H \times H$
2. $|A \times B| = |A||B|$

where the equality on G/H is defined by $\hat{x} = \hat{y}$ iff $xy^{-1} \in H$. We do not always have:

$$\forall a : A \exists b : B, (Rab) \rightarrow \exists f : (A \Rightarrow B), \forall a : A, (Ra(fa))$$

but we can prove it for finite A.

The isomorphism is given by:

$$f(g) = \langle \hat{g}, g \cdot (c(g))^{-1} \rangle$$
$$f^{-1}\langle \hat{g}, h \rangle = h \cdot (c(g))$$

(2) is proved by induction over the size of B.

What I had and what needed to be done

Corn has:

- Setoids, including functions, union and products between setoids.
- Groups, including simplification of algebraic expressions.

I had to do:

- Define finiteness.
- Define isomorphisms and prove lemmas about it.

  $\simeq$ forms equivalence, $A \simeq (S \mid P) \cup (A \mid \neg P)$ and

  $A \times (B \cup C) \simeq A \times B \cup A \times C$ about 15 more
- Relate decidability to isomorphisms, finiteness and subsets.
- Define extensional choice and prove it for finite sets.

So what is the problem with Setoids

A Setoid is a type and an equivalence relation.

- No obvous choice how to represent set operations. Should one work with predicates representing sets, with union as predicate operation, or work with disjoint union on the carriers? Is a subgroup and group based on different notion of set?
- Intensionality of type theory together with setoids often forces you to use isomorphisms, e.g. subset and quotient don't commute.
- A common construction is to make a setoid based on a sigma type with the carrier and a predicate, with the equality "inherited" from the original setoid. In practice using these setoids leads to extensive packing and unpacking (as observed by Luis Cruz-Filipe).
- $A \mid P \mid Q \simeq A \mid Q \mid P$ is not a meaningful statement.

Some observations

- Around 6000 lines of proof.
- Most lines are spent on proving lemmas about isomorphisms.
- We could hide the size of the proof behind fancy tactics. But is that what we want?
- Things like better syntax, setoid rewriting and a big library is not the solution.

Future Work

- Formalising can be a nice way to illustrate the power of abstraction. Motivation for proof assistants as a tool in teaching.
- Is PER/subsetoids a better framework to represent sets?
- Or more radically: Work in ZF coded in type theory a la Miguel
- Or even more radically: Find an application that justifies the choice of framework!