

Something to Lean on; fun with interactive theorem provers

Something to Lean on; fun with interactive theorem provers

Alco J. Best
BU Math Retreat 2019

1. Thank the audience for being awake.

2019-04-29

Something to Lean on; fun with interactive theorem provers

└ The problem

like a living organism mathematics notices and fixes itself eventually (this is harder for a false proof of a true statement!).

The problem

Mathematicians make mistakes.

Sometimes they publish these mistakes.

Sometimes nobody notices.

At least for a while...

However this uncertainty takes up time and energy, what if referees only needed to judge the importance, novelty and quality of exposition, not check the arguments.

Something to Lean on; fun with interactive theorem provers

└ Some examples: Grunwald(-Wang) and K -theory

Some examples: Grunwald(-Wang) and K -theory

Some days later I was with Artin in his office when Wang appeared. He said he had a counterexample to a lemma which had been used in the proof. An hour or two later, he produced a counterexample to the theorem itself. . . Of course he [Artin] was astonished, as were all of us students, that a famous theorem with two published proofs, one of which we had all heard in the seminar without our noticing anything, could be wrong.

— Tate

The groundbreaking 1986 paper "Algebraic Cycles and Higher K -theory" by Spencer Bloch was soon after publication found by Andrei Suslin to contain a mistake in the proof of Lemma 1.1. The proof could not be fixed.

— Voevodsky

The problem: 2, the cursed prime. This is often an edge case.

2019-04-29

Something to Lean on; fun with interactive theorem provers

└ The new problem

(this is one place mistakes enter) Teach the computer to work off as little as possible.

The new problem

How do you describe the steps of a proof to a computer with as little pain as possible? Often mathematicians leave unsaid many steps which are intuitive or easily supplied.

Proof: (left as an exercise) (Hint: Use Binet's formula).

Bevezir. Klar.

[Proof similar to that of #3531]

Proof Obvious.

Proof: Left as an exercise. (Hint: do you remember DeMorgan's Laws?)

Proof: Easy, by induction on n .

Q. E. D.

The computer will probably not understand these, but in order to stay sane we must strike a balance between detail and verbosity.

Something to Lean on; fun with interactive theorem provers

```

lemma fact_rec (n : ℕ) :
  factorial (n + 1) = factorial n * (n+1) :=
begin
  -- write out the definition of factorial
  unfold factorial,
  -- remember {1,...,n+1} = {1,...,n} ∪ {n+1}
  rewrite list.range'_concat 1 n,
  -- the product of two sequences joined together is
  -- just the product of the products of each sequence
  rewrite list.prod_append,
  -- I'm bored already are we done here?
  simp,
  -- YES!
end

We can replace all of the above with: by unfold factorial;
simp [list.range'_concat, list.prod_append]

```

Lean will figure out when and how to apply the lemmas.