Sledgehammering Without ATPs

Martin Desharnais-Schäfer
with Jasmin Blanchette

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

16th International Conference on Interactive Theorem Proving (ITP 2025)

Proof method

Try to automatically solve a proof goal using different algorithm or heuristics.

Examples `simp`, `auto`, `blast`, `metis`, …

Proof Automation
Tools in Isabelle/HOL

Proof method

Try to automatically solve a proof goal using different algorithm or heuristics.

Examples `simp`, `auto`, `blast`, `metis`, ...

`try0`

Push-button automation using **internal proof methods**.

Proof Automation
Tools in Isabelle/HOL

Proof method

Try to automatically solve a proof goal using different algorithm or heuristics.

Examples `simp`, `auto`, `blast`, `metis`, …

`try0`

Push-button automation using **internal proof methods**.

Sledgehammer

Push-button automation using **external automatic theorem provers (ATPs)**.

Components Relevance filter, translation module, external ATPs, proof minimization, proof reconstruction

Proof Automation

User Perspective

Automation-driven development

1. State the proof goal.
2. Try automation (e.g., `try0` or Sledgehammer).
3. Stop if proof found;
   otherwise, find an intermediate lemma to help automation and go to step 1.

Proof Automation
User Perspective

Automation-driven development

1. State the proof goal.
2. Try automation (e.g., `try0` or Sledgehammer).
3. Stop if proof found;
   otherwise, find an intermediate lemma to help automation and go to step 1.

Caveats:

- `try0` and Sledgehammer have different strengths: Manually **call both**.
- `try0` has no fact filter: Manually **provide facts**.

ATP-Free Hammer

We propose an alternative, ATP-free hammer:

1. **Relevance filter**: as before
2. **Translation module**: not necessary
3. **External ATPs**: replaced by internal proof methods
4. **Proof minimization**: as before
5. **Proof reconstruction**: not necessary

Motivating Example

```
have map2 (+) (map2 (+) xs ys) zs = map2 (+) xs (map2 (+) ys zs)
  for xs ys zs :: (‘a :: ab_semigroup_add) list
```

Motivating Example

have map2 $(+)$ (map2 $(+)$ *xs* *ys*) *zs* $=$ map2 $(+)$ *xs* (map2 $(+)$ *ys* *zs*)
  for *xs* *ys* *zs* :: ('*a* :: ab_semigroup_add) list

Automation-driven development:

1. `try0`: Timeout, no proof found.
2. ATP hammer: Timeout, no proof found.

Motivating Example

have map2 (+) (map2 (+) *xs ys*) *zs* = map2 (+) *xs* (map2 (+) *ys zs*)
for *xs ys zs* :: ('*a* :: ab_semigroup_add) list

Automation-driven development:

1. `try0`: Timeout, no proof found.
2. ATP hammer: Timeout, no proof found.
3. ATP-free hammer: Proof found!

by (simp add: case_prodI2 prod.case_distrib zip_assoc case_prod_app map_zip_map
map_zip_map2 ab_semigroup_add_class.add_ac(1))

Experiment
Methodology

Benchmark

We randomly selected **20 formalizations** and, in each, **250 goals**: a total of **5000 tests cases**.

Experiment
Methodology

Benchmark

We randomly selected **20 formalizations** and, in each, **250 goals**: a total of **5000 tests cases**.

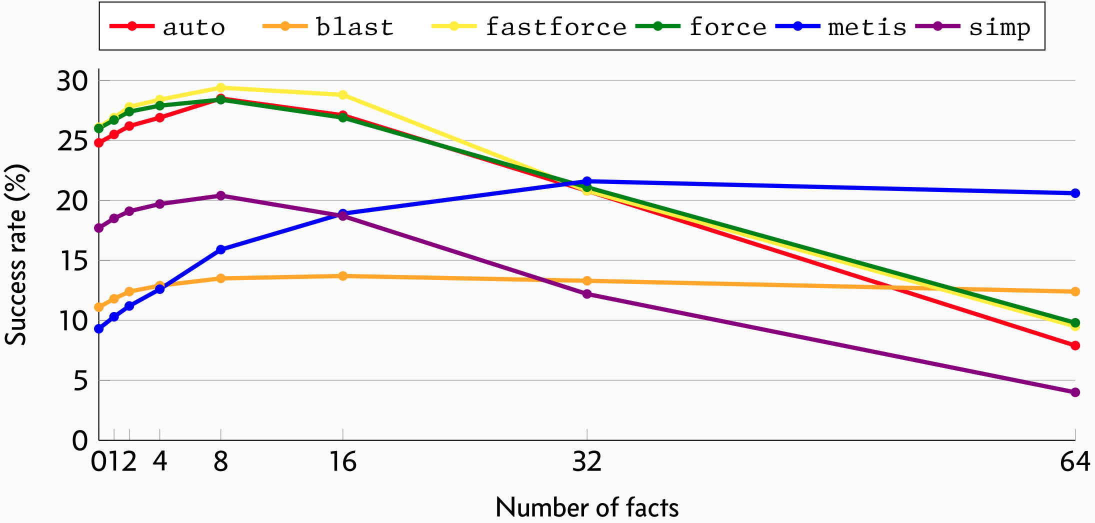
Configurations

We tested **13 proof methods** with a **2-second timeout** and between **0 and 64 facts**.

The 13 proof methods with 0 facts correspond to `try0`.

Experiment

Detailed Results (Selected Proof Methods)



Experiment

Overall Results

Configurations	Success rate (%)
`try0`	28.5
ATP hammer	72.1
ATP-free hammer	46.8
ATP hammer ∪ `try0`	74.1
ATP hammer ∪ ATP-free hammer	74.6

Remainder: `try0` ⊆ ATP-free hammer

Conclusion: `try0` and ATP-free hammer both improve the ATP hammer!

Discussion and Conclusion

We implemented and evaluated an **ATP-free hammer** in Isabelle/HOL.
It can find **new proofs** that the ATP hammer cannot.

We **extended Sledgehammer** with `try0` and ATP-free portfolio.

Discussion and Conclusion

We implemented and evaluated an **ATP-free hammer** in Isabelle/HOL.
It can find **new proofs** that the ATP hammer cannot.

We **extended Sledgehammer** with `try0` and ATP-free portfolio.

Take-home message

Isabelle/HOL users can now use only Sledgehammer:

- No need to manually call `try0`
- Expect better results

Discussion and Conclusion

We implemented and evaluated an **ATP-free hammer** in Isabelle/HOL.
It can find **new proofs** that the ATP hammer cannot.

We **extended Sledgehammer** with `try0` and ATP-free portfolio.

Take-home message

Isabelle/HOL users can now use only Sledgehammer:

- No need to manually call `try0`
- Expect better results

Thank you!