Automated Theorem Proving = Logic Programming

… and about resolution

@lambduli

So We Have Logic

That’s great, isn’t it?

We can reason about stuff.

First Order Logic

The one that's mostly enough

When it's not, second-order should be fine … mostly … hopefully … usually.

What about induction?

Logic Programming

Is it really about programming?

Isn’t it, perhaps, about automated theorem proving?

Or at least—about automated theorem proving ***too***?

Prolog

Prolog, even though not the full FOL, can be see as an automated theorem prover of sorts.

Not a good one, though.

See https://www.metalevel.at/prolog/theoremproving

Prolog is a Subset of FOL

Horn Clauses

A **Horn clause** is a disjunctive clause (a disjunction of literals) with at most one positive, i.e. unnegated, literal.

$$A \lor \neg B \lor \neg C \lor \neg D$$

$$B \land C \land D \implies A$$

```
A :- B, C, D
```

```
plus(zero, X, X).

plus(suc(X), Y, suc(Z)) :- plus(X, Y, Z).

?- plus(A, B, A).
```

$\forall\ x\ \top \implies Plus(Zero, x, x)$

$\forall\ x\ y\ z\ Plus(x, y, z) \implies Plus(suc(x), y, suc(z))$

$\exists\ a\ b\ Plus(a, b, a) \implies \bot$

$\forall\ x\ \neg\top \lor Plus(Zero, x, x)$

$\forall\ x\ y\ z\ \neg Plus(x, y, z) \lor Plus(suc(x), y, suc(z))$

$\forall\ a\ b\ \neg Plus(a, b, a) \lor \neg\top$

Horn Clauses

They Are Great

Deciding entailment with Horn clauses can be done in time that is linear in the size of the knowledge base.

What If That's not Enough?

We can't use Prolog (or Prolog with complete search strategy).

We use *resolution.*

Propositional Resolution

Much Simpler than FOL Resolution

$A \lor B \lor \neg C \quad \land \quad C \lor D \lor E$

$A \lor B \lor D \lor E$

Resolution

We prove validity of statements.

It’s refutation complete.

It’s really great for a proof by contradiction of entailment statements.

$A \lor B \land C \vdash D$

Refutation Completeness

I don’t Care about Gödel!

If the “information is there” resolution can prove it.

In other words, if the RHS is indeed a logical consequence of the LHS we negate it and “conjugate” it with the LHS and let the resolution have at it.

If there’s a contradiction, resolution will derive it in a finite amount of steps.

When that happens, it means that the negation is unsatisfiable therefore the original statement is logically valid.

Demo