System Description: GAPT 2.0

Gabriel Ebner Stefan Hetzl Giselle Reis Martin Riener
Simon Wolfsteiner Sebastian Zivota

June 27, 2016

1. Motivation

2. Features

3. Demonstration

4. Future work

Motivation

- Computational proof theory
    - Classical first-/higher-order logic
    - Cut-elimination
    - Herbrand disjunctions
    - ...

⇒ Implementation in a common framework: GAPT

- "General Architecture for Proof Theory"
- Scala library

Applications

Lemma generation

- Example: transitivity in meet-join-lattices.
- Requires:
  - First-order (expansion) proofs
  - SAT/SMT solvers
  - MaxSAT solvers

Cut-elimination by resolution

- Resolution refutation as skeleton of cut-free proof
- Requires:
  - Proof input
  - Resolution proofs
  - Conversion to expansion proofs
  - Visualization

1 Motivation

2 Features

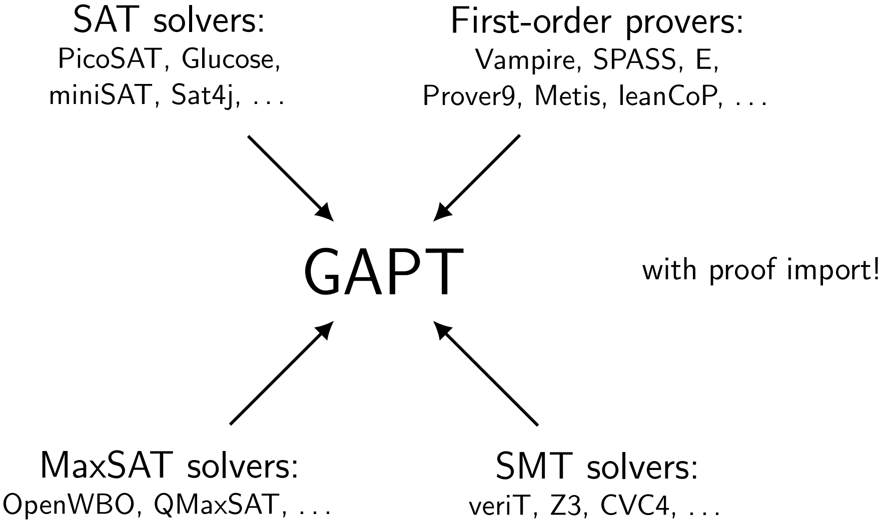
3 Demonstration

4 Future work

Basic functionality

- Formulas, positions, substitution, unification, subsumption, . . .
- Parsing
- Export
- Clausification
- . . .

Interfaces to external solvers



SAT solvers:
PicoSAT, Glucose,
miniSAT, Sat4j, . . .

First-order provers:
Vampire, SPASS, E,
Prover9, Metis, leanCoP, . . .

GAPT

with proof import!

MaxSAT solvers:
OpenWBO, QMaxSAT, . . .

SMT solvers:
veriT, Z3, CVC4, . . .

Proof systems

⇒ Classical higher-order logic with equality

- Expansion proofs
  - ≃ generalized Herbrand-disjunctions
  - answer substitutions / witness terms
- Sequent calculus (LK)
- Resolution proofs
- ...

Operations on proofs

- Conversions
- Tactics (for sequent calculus)
  - Embedded domain-specific language
- Visualization
- Transformations
  - Cut-elimination
  - Minimization
- and more!

Demonstration

1. Motivation

2. Features

3. Demonstration

4. Future work

Future work

- Inductive prover based on tree grammars

- Synthesis by program extraction
- Natural deduction

- Deskolemization

- New solver interfaces: QBF, more SMT theories, . . .

- `https://logic.at/gapt`