Set of Support for Theory Reasoning

Giles Reger[1], Martin Suda[2]

[1]School of Computer Science, University of Manchester, UK
[2]TU Wien, Vienna, Austria

IWIL 2017 – Maun, May 7, 2017

Theory axioms in proofs

Consider the following toy theory problem

$$f(1 + a) < a, \qquad \forall x.(x < f(x + 1))$$

Theory axioms in proofs

Consider the following toy theory problem

$$f(1+a) < a, \qquad \forall x.(x < f(x+1))$$

can be refuted by Vampire via the following derivation:

$$\dfrac{\dfrac{\dfrac{x+y=y+x \qquad x < f(x+1)}{x < f(1+x)} \qquad \dfrac{\neg x < y \vee \neg y < z \vee x < z \qquad f(1+a) < a}{\neg(x < f(1+a)) \vee x < a}}{a < a} \qquad \neg(x < x)}{\bot}$$

Theory axioms in proofs

Consider the following toy theory problem

$$f(1+a) < a, \qquad \forall x.(x < f(x+1))$$

can be refuted by Vampire via the following derivation:

$$\dfrac{\dfrac{\dfrac{x+y=y+x \qquad x<f(x+1)}{x<f(1+x)} \qquad \dfrac{\neg x<y \vee \neg y<z \vee x<z \qquad f(1+a)<a}{\neg(x<f(1+a)) \vee x<a}}{a<a} \qquad \neg(x<x)}{\bot}$$

However, in the meantime, the theory axioms may also yield:

$$\neg(x<y) \vee \neg(y<x)$$

or (perhaps less usefully):

$$\neg(x_0<x_1) \vee \neg(x_2<x_0) \vee \neg(x_1<x_3) \vee \neg(x_4<x_5) \vee \neg(x_3<x_4) \vee \neg(x_5<x_2)$$

Inferences between axioms

Example problem ARI176=1 from TPTP

$$3x + 5y \neq 22$$

can be shown unsatisfiable using axioms

$$x{+}y = y{+}x, \quad x{+}(y{+}z) = (x{+}y){+}z, \quad x{*}1 = x, \quad x{*}(y{+}z) = (x{*}y){+}(x{*}z)$$

Inferences between axioms

Example problem ARI176=1 from TPTP

$$3x + 5y \neq 22$$

can be shown unsatisfiable using axioms

$$x{+}y = y{+}x, \quad x{+}(y{+}z) = (x{+}y){+}z, \quad x{*}1 = x, \quad x{*}(y{+}z) = (x{*}y){+}(x{*}z)$$

The derivation starts by:

$$\dfrac{\dfrac{x * 1 = x \qquad x * (y + z) = (x * y) + (x * z)}{x * (1 + y) = x + (x * y)} \qquad x + (y + z) = (x + y) + z}{(x * (1 + y)) + z = x + ((x * y) + z)}$$

Inferences between axioms

Example problem `ARI176=1` from TPTP

$$3x + 5y \neq 22$$

can be shown unsatisfiable using axioms

$x{+}y = y{+}x, \quad x{+}(y{+}z) = (x{+}y){+}z, \quad x{*}1 = x, \quad x{*}(y{+}z) = (x{*}y){+}(x{*}z)$

The derivation starts by:

$$\dfrac{\dfrac{x * 1 = x \qquad x * (y + z) = (x * y) + (x * z)}{x * (1 + y) = x + (x * y)} \qquad x + (y + z) = (x + y) + z}{(x * (1 + y)) + z = x + ((x * y) + z)}$$

The problem cannot be solved in Vampire in reasonable time without first combining axioms among themselves

This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms

This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms
- Quite successful in many cases.
  However, many axioms can be "explosive".

This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms
- Quite successful in many cases.
  However, many axioms can be "explosive".
- Set of support is a well known idea to prevent explosion

This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms
- Quite successful in many cases.
  However, many axioms can be "explosive".
- Set of support is a well known idea to prevent explosion
- Idea 1: apply SOS for theory reasoning

This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms
- Quite successful in many cases.
  However, many axioms can be "explosive".
- Set of support is a well known idea to prevent explosion
- Idea 1: apply SOS for theory reasoning
- Idea 2: fine-tune this by allowing limited reasoning among theory axioms

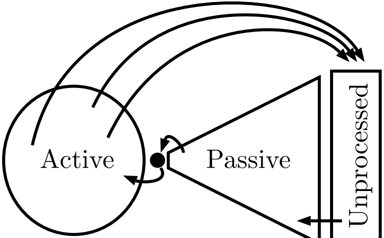
This talk in a nutshell

- One useful technique for reasoning with theories and quantifiers is the addition of theory axioms
- Quite successful in many cases.
  However, many axioms can be "explosive".
- Set of support is a well known idea to prevent explosion
- Idea 1: apply SOS for theory reasoning
- Idea 2: fine-tune this by allowing limited reasoning among theory axioms
- Preliminary evaluation of the technique

Outline

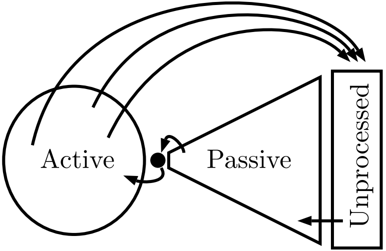
1. Saturation and Theory Reasoning in Vampire
2. The Set of Support Strategy
3. Set of Support for Theory Reasoning
4. Conclusion

Compute deductive closure of the input N wrt inferences $\mathcal{I}$:



Saturation-based Theorem Proving

Compute deductive closure of the input N wrt inferences $\mathcal{I}$:



- clause selection schemes
- further aspects: literal selection, ordering restrictions, ...
- completeness considerations

Main focus

Reasoning with quantifiers and theories

Main focus

Reasoning with quantifiers and theories

Current arsenal:

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow \mathit{false}$, ...

Theory Reasoning in Vampire

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow \mathit{false}$, ...
- Interpreted operations treated specially by ordering

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow$ *false*, ...
- Interpreted operations treated specially by ordering
- Normalization of interpreted operations, i.e. only use $<$

Theory Reasoning in Vampire

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow \mathit{false}$, ...
- Interpreted operations treated specially by ordering
- Normalization of interpreted operations, i.e. only use $<$
- Theory axioms
  - hand-crafted set
  - either all added or none added (based on an option)

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow$ *false*, ...
- Interpreted operations treated specially by ordering
- Normalization of interpreted operations, i.e. only use $<$
- Theory axioms
  - hand-crafted set
  - either all added or none added (based on an option)
- AVATAR with an SMT solver
  - Idea: Vampire only explores theory-consistent ground sub-problems

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow$ *false*, ...
- Interpreted operations treated specially by ordering
- Normalization of interpreted operations, i.e. only use $<$
- Theory axioms
  - hand-crafted set
  - either all added or none added (based on an option)
- AVATAR with an SMT solver
  - Idea: Vampire only explores theory-consistent ground sub-problems
- Theory Instantiation and Unification with Abstraction

Main focus

Reasoning with quantifiers and theories

Current arsenal:

- Evaluation of ground interpreted terms:
  $1 + 1 \Longrightarrow 2$, $1 < 1 \Longrightarrow \mathit{false}$, ...
- Interpreted operations treated specially by ordering
- Normalization of interpreted operations, i.e. only use $<$
- Theory axioms
  - hand-crafted set
  - either all added or none added (based on an option)
- AVATAR with an SMT solver
  - Idea: Vampire only explores theory-consistent ground sub-problems
- Theory Instantiation and Unification with Abstraction

Some axioms

$$x + (y + z) = (x + y) + z \qquad x + 0 = x$$
$$x + y = y + x \qquad -(x + y) = (-x + -y)$$
$$- - x = x \qquad x + (-x) = 0$$
$$x * 0 = 0 \qquad x * (y * z) = (x * y) * z$$
$$x * 1 = x \qquad x * y = y * x$$
$$(x * y) + (x * z) = x * (y + z) \qquad \neg(x < y) \vee \neg(y < z) \vee \neg(x < z)$$
$$x < y \vee y < x \vee x = y \qquad \neg(x < y) \vee \neg(y < x + 1)$$
$$\neg(x < y) \vee x + z < y + z \qquad \neg(x < x)$$
$$x < y \vee y < x + 1 \ (for\ ints) \qquad x = 0 \vee (y * x)/x = y \ (for\ reals)$$

Axioms can be "explosive"

ARI581=1.p

```
tff(mix_quant_ineq_sys_solvable_2,conjecture,(
  ! [X: $int] : ( $less(5,X) =>
   ? [Y: $int] : ( $less(Y,3) & $less(7,$sum(X,Y)))))).
```

- default strategy with all axioms: not solved in 60 s
- remove commutativity of $+$: solved instantly

Axioms can be “explosive”

ARI581=1.p

```
tff(mix_quant_ineq_sys_solvable_2,conjecture,(
  ! [X: $int] : ( $less(5,X) =>
   ? [Y: $int] : ( $less(Y,3) & $less(7,$sum(X,Y)))))).
```

- default strategy with all axioms: not solved in 60 s
- remove commutativity of $+$: solved instantly

SYN000=2.p

- “test tptp theory syntax” benchmark
- Vampire in default: 223 clauses (90 theory consequences, 1 used in the proof)
- negate the conjecture, run for 10 s:
  456 973 clauses (98 % are consequences of theory axioms)

Outline

1 Saturation and Theory Reasoning in Vampire

2 The Set of Support Strategy

3 Set of Support for Theory Reasoning

4 Conclusion

The Set of Support Strategy

Basic idea:

- split the input clauses into a set of support and the rest
- restrict inferences to involve at least one premise from SOS
- new clauses are added to SOS

"Every inference must have an ancestor in the initial SOS."

The Set of Support Strategy

Basic idea:

- split the input clauses into a <u>set of support</u> and the rest
- restrict inferences to involve at least one premise from SOS
- new clauses are added to SOS

"Every inference must have an ancestor in the initial SOS."

In practice:

- just put non-SOS clauses directly to active

The Set of Support Strategy

Basic idea:

- split the input clauses into a set of support and the rest
- restrict inferences to involve at least one premise from SOS
- new clauses are added to SOS

"Every inference must have an ancestor in the initial SOS."

In practice:

- just put non-SOS clauses directly to active
- define SOS = clauses from the conjecture
  - Note: benchmarks without explicit conjecture SOS-suck

SOS in Vampire

Vampire's `-sos` option values:

- `off`: do not use SOS
- `on`: standard SOS
- `all`: SOS + select all literals of clauses in "initially active"

SOS in Vampire

Vampire's `-sos` option values:

- `off`: do not use SOS
- `on`: standard SOS
- `all`: SOS + select all literals of clauses in “initially active”

Experiment (relevant TPTP v6.4.0, 300 s)

	competition mode	competition mode with `sos=off`
Solved	11 948	11 613
Uniques	422	87

Outline

1. Saturation and Theory Reasoning in Vampire
2. The Set of Support Strategy
3. Set of Support for Theory Reasoning
4. Conclusion

SOS for Theories

SOS and theory axioms

- the whole input problem is the SOS
- added theory axioms go directly to active
- new, fourth `-sos` option value: `theory`

SOS for Theories

SOS and theory axioms

- the whole input problem is the SOS
- added theory axioms go directly to active
- new, fourth `-sos` option value: `theory`
- Also applies to problems without explicit conjecture!

SOS for Theories

SOS and theory axioms

- the whole input problem is the SOS
- added theory axioms go directly to active
- new, fourth `-sos` option value: `theory`
- Also applies to problems without explicit conjecture!

Experiment (relevant SMTLIB, default strategy, 60 s)

	default mode	default mode + `sos=theory`
Solved	32 769	32 522
Uniques	641	394

How deep is theory reasoning?

Mining proofs for statistics:

- record maximum derivation depth
  of a pure theory consequence used in the proof

How deep is theory reasoning?

Mining proofs for statistics:

- record maximum derivation depth
  of a pure theory consequence used in the proof

Experiment (relevant SMTLIB, default strategy, 60 s)

Depth	count
0	31 959
1	209
2	304
3	200
4	49
5	21
6	27

What do useful pure theory consequences look like?

Example (deep pure theory consequences)

$$0 < x \vee x < 4$$

from `UFLIA/sledgehammer/TwoSquares/z3.637729.smt2`

$$\neg((x + (y + ((-x) + 2.0))) < y) \quad \text{and} \quad \neg(2.0 + x < x)$$

from `NRA/keymaera/ETCS-essentials-live-range2.proof-node1388.smt2`

What do useful pure theory consequences look like?

Example (deep pure theory consequences)

$$0 < x \vee x < 4$$

from `UFLIA/sledgehammer/TwoSquares/z3.637729.smt2`

$$\neg((x + (y + ((-x) + 2.0))) < y) \quad \text{and} \quad \neg(2.0 + x < x)$$

from `NRA/keymaera/ETCS-essentials-live-range2.proof-node1388.smt2`

Note that:

- large constants must be obtained by combining the basic axioms

What do useful pure theory consequences look like?

Example (deep pure theory consequences)

$$0 < x \vee x < 4$$

from `UFLIA/sledgehammer/TwoSquares/z3.637729.smt2`

$$\neg((x + (y + ((-x) + 2.0))) < y) \quad \text{and} \quad \neg(2.0 + x < x)$$

from `NRA/keymaera/ETCS-essentials-live-range2.proof-node1388.smt2`

Note that:

- large constants must be obtained by combining the basic axioms
- a clumsy search for a useful instance?

Explicitly liming depth of pure theory consequences

Depth	Count when threshold = 0	1	2	3	5	10	∞
0	32 522	32 253	32 130	32 061	32 162	32 040	31 959
1		552	237	209	216	208	209
2			551	314	310	307	304
3				312	254	212	200
4					69	48	49
5					61	21	21
6						26	27
total	32 522	32 805	32 918	32 896	**33 072**	32 863	32 769

Some further observations

Let us denote the depth threshold T

- solved with $T = n$ can still be solvable with $T = m < n$

Some further observations

Let us denote the depth threshold T

- solved with $T = n$ can still be solvable with $T = m < n$
- decreasing T can dramatically decrease the solution time and length of the found proof

Some further observations

Let us denote the depth threshold T

- solved with $T = n$ can still be solvable with $T = m < n$
- decreasing T can dramatically decrease the solution time and length of the found proof
- decreasing T can also dramatically increase the solution time and length of the found proof

Some further observations

Let us denote the depth threshold T

- solved with $T = n$ can still be solvable with $T = m < n$
- decreasing T can dramatically <u>decrease</u> the solution time and length of the found proof
- decreasing T can also dramatically <u>increase</u> the solution time and length of the found proof

Experiment (relevant SMTLIB, smtcomp mode, 1800 s)

	competition mode	`set sos=theory threshold=5`
Solved	37 009	36 821
Uniques	254	66

Conclusion

Summary

- adapted SOS for dealing with theory axioms
- tuned by a derivation depth parameter
- promising initial experiments

Conclusion

Summary

- adapted SOS for dealing with theory axioms
- tuned by a derivation depth parameter
- promising initial experiments

Ideas and plans for future work:

- better understand relations to other theory reasoning techniques

Conclusion

Summary

- adapted SOS for dealing with theory axioms
- tuned by a derivation depth parameter
- promising initial experiments

Ideas and plans for future work:

- better understand relations to other theory reasoning techniques
- what are the useful (deep) theory consequences?
  could they be precomputed?

Conclusion

Summary

- adapted SOS for dealing with theory axioms
- tuned by a derivation depth parameter
- promising initial experiments

Ideas and plans for future work:

- better understand relations to other theory reasoning techniques
- what are the useful (deep) theory consequences? could they be precomputed?
- distinguish "explosiveness" of axioms on case by case basis

Conclusion

Summary

- adapted SOS for dealing with theory axioms
- tuned by a derivation depth parameter
- promising initial experiments

Ideas and plans for future work:

- better understand relations to other theory reasoning techniques
- what are the useful (deep) theory consequences?
  could they be precomputed?
- distinguish "explosiveness" of axioms on case by case basis

Thank you for your attention!