Cooperation between SAT, SMT provers and Coq

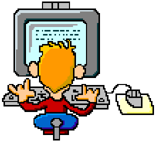
Michaël Armand Germain Faure Benjamin Grégoire
Chantal Keller Laurent Théry Benjamin Werner

INRIA – École Polytechnique

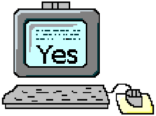
April, 2[nd] 2011

Introduction

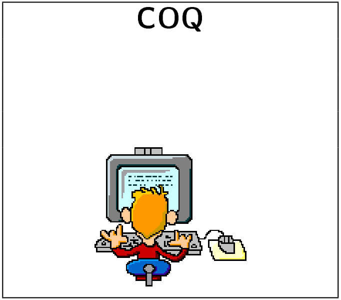
COQ

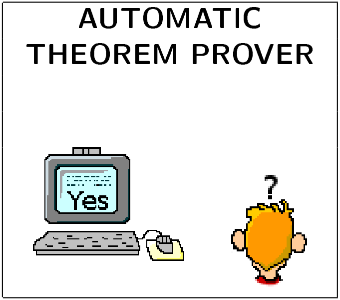


AUTOMATIC THEOREM PROVER

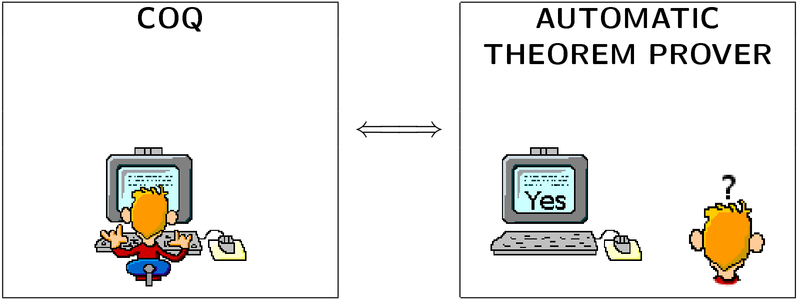


Introduction



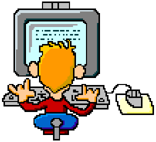


Introduction



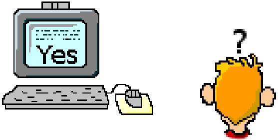
Introduction

COQ

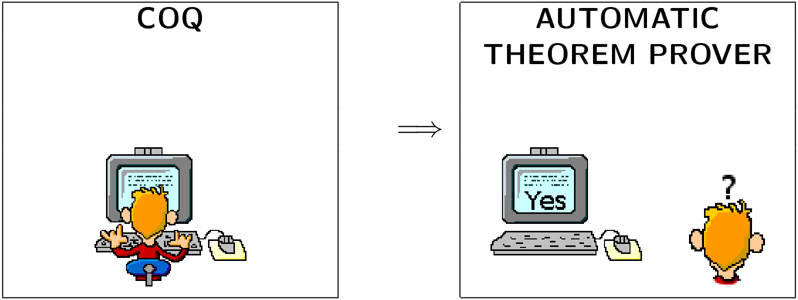


$\Longleftarrow$

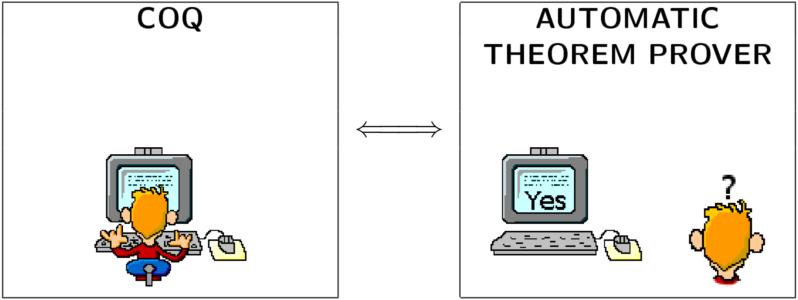
AUTOMATIC THEOREM PROVER



Introduction



Introduction



Outline

Proof producing SAT solvers

Decide propositional satisfiability of sets of clauses:

- $x \vee y \qquad x \vee \bar{y} \vee z \qquad \bar{x} \vee z \qquad \bar{z}$

Proof witness:

- If satisfiable: assignment of the variables to $\top$ or $\bot$
- If unsatisfiable: proof by resolution of the empty clause

Resolution rule:

$$\frac{x \vee C \qquad \bar{x} \vee D}{C \vee D}$$

Examples

Satisfiability of: $x \vee y \qquad x \vee \bar{y} \vee z \qquad \bar{x} \vee z$

$$\{x \mapsto \top, y \mapsto \bot, z \mapsto \top\}$$

Unsatisfiability of: $x \vee y \qquad x \vee \bar{y} \vee z \qquad \bar{x} \vee z \qquad \bar{z}$

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{\square}$$

Examples

Satisfiability of: $x \vee y \quad x \vee \bar{y} \vee z \quad \bar{x} \vee z$

$$\{x \mapsto \top, y \mapsto \bot, z \mapsto \top\}$$

Unsatisfiability of: $x \vee y \quad x \vee \bar{y} \vee z \quad \bar{x} \vee z \quad \bar{z}$

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{\square}$$

Proof producing SMT solvers

Atoms are now formulas of some theories:

- congruence closure
- linear arithmetic
- ...
- $f(x) \neq f(y) \qquad f(x) = f(f(z)) \qquad x = y$

Proof witness:

- If satisfiable: assignment of the variables
- If unsatisfiable: proof by resolution of the empty clause in which some leaves are theory lemmas

Examples

Satisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z))$

$$\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$$

Unsatisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z)) \qquad x = y$

$$\dfrac{\dfrac{x \neq y \vee f(x) = f(y) \qquad x = y}{f(x) = f(y)} \qquad f(x) \neq f(y)}{\Box}$$

Examples

Satisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z))$

$$\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$$

Unsatisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z)) \qquad x = y$

$$\dfrac{\dfrac{x \neq y \vee f(x) = f(y) \qquad x = y}{f(x) = f(y)} \qquad f(x) \neq f(y)}{\square}$$

Examples

Satisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z))$

$$\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$$

Unsatisfiability of: $f(x) \neq f(y) \qquad f(x) = f(f(z)) \qquad x = y$

$$\dfrac{\dfrac{x \neq y \vee f(x) = f(y) \qquad x = y}{f(x) = f(y)} \qquad f(x) \neq f(y)}{\square}$$

Such tools

SAT solvers

MiniSat

zChaff

SMT solvers

Z3

veriT

CVC3

Coq: programming language **and** interactive theorem prover

A programming language:

- We can write a checker of proof witnesses

An interactive theorem prover:

- We can prove the correctness of this checker

A proof involving automatic provers is thus:

- An application of the correctness lemma
- Computation of the checker

$\hookrightarrow$ Very small proof terms (proof by reflection)

Coq: programming language **and** interactive theorem prover

A programming language:

- We can write a checker of proof witnesses
- And even export it to ML

An interactive theorem prover:

- We can prove the correctness of this checker

A proof involving automatic provers is thus:

- An application of the correctness lemma
- Computation of the checker

$\hookrightarrow$ Very small proof terms (proof by reflection)

... with efficient computation

Native data structures:

- arrays
- machine integers

Conclusion:

- Small proof terms with fast computation

Outline

SAT and UNSAT

Main idea:

- SAT: replace the variables by their assignments and check the result is $\top$
- UNSAT: check the resolution tree

Checking the resolution tree implies:

- checking resolution steps
- checking theory lemmas
- interface between them

SAT and UNSAT

Main idea:

- SAT: replace the variables by their assignments and check the result is $\top \hookrightarrow$ easy
- UNSAT: check the resolution tree

Checking the resolution tree implies:

- checking resolution steps
- checking theory lemmas
- interface between them

SAT and UNSAT

Main idea:

- SAT: replace the variables by their assignments and check the result is $\top$ $\hookrightarrow$ easy
- UNSAT: check the resolution tree $\hookrightarrow$ more difficult

Checking the resolution tree implies:

- checking resolution steps
- checking theory lemmas
- interface between them

A modular checker

Given "small" checkers:

- a resolution checker
- a checker for each theory

The "interface" checker:

- checks the resolution tree
- at each step, calls one of the small checkers

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{\square}$$

The "interface" checker by Example

$$\frac{\dfrac{x \lor y \qquad \dfrac{x \lor \bar{y} \lor z \qquad \bar{z}}{x \lor \bar{y}}}{x} \qquad \dfrac{\bar{x} \lor z \qquad \bar{z}}{\bar{x}}}{\Box}$$

8 different clauses:

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{}}{} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{}}{}$$

The "interface" checker by Example

$$\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{}}{}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$				
↑	↑	↑	↑				

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{}}{}$$

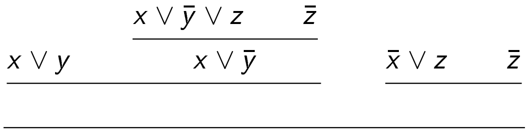
$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$				

The "interface" checker by Example

$$\dfrac{x \lor y \qquad \dfrac{x \lor \bar{y} \lor z \qquad \bar{z}}{x \lor \bar{y}} \qquad \dfrac{\bar{x} \lor z \qquad \bar{z}}{}}{}$$

$x \lor y$	$x \lor \bar{y} \lor z$	$\bar{z}$	$\bar{x} \lor z$				

The "interface" checker by Example



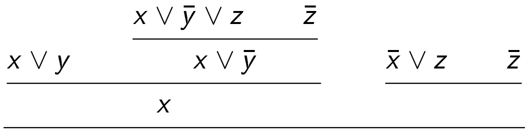
$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$			

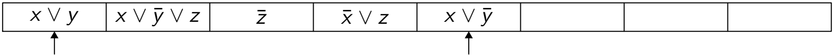
The "interface" checker by Example

$$\dfrac{\dfrac{x \lor y \qquad \dfrac{x \lor \bar{y} \lor z \qquad \bar{z}}{x \lor \bar{y}}}{x} \qquad \dfrac{\bar{x} \lor z \qquad \bar{z}}{}}{}$$

$x \lor y$	$x \lor \bar{y} \lor z$	$\bar{z}$	$\bar{x} \lor z$	$x \lor \bar{y}$			

The "interface" checker by Example





The "interface" checker by Example

$$\cfrac{x \vee y \qquad \cfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \cfrac{\bar{x} \vee z \qquad \bar{z}}{}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$	x		

↑ (at x)

The "interface" checker by Example

$$\dfrac{\dfrac{x \lor y \qquad \dfrac{x \lor \bar{y} \lor z \qquad \bar{z}}{x \lor \bar{y}}}{x} \qquad \dfrac{\bar{x} \lor z \qquad \bar{z}}{\bar{x}}}{}$$

$x \lor y$	$x \lor \bar{y} \lor z$	$\bar{z}$	$\bar{x} \lor z$	$x \lor \bar{y}$	x		

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$	x		
		↑	↑				

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$	x	$\bar{x}$	
						$\uparrow$	

The "interface" checker by Example

$$\dfrac{\dfrac{x \lor y \qquad \dfrac{x \lor \bar{y} \lor z \qquad \bar{z}}{x \lor \bar{y}}}{x} \qquad \dfrac{\bar{x} \lor z \qquad \bar{z}}{\bar{x}}}{\square}$$

$x \lor y$	$x \lor \bar{y} \lor z$	$\bar{z}$	$\bar{x} \lor z$	$x \lor \bar{y}$	x	$\bar{x}$	

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{\square}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$	x	$\bar{x}$	
					↑	↑	

The "interface" checker by Example

$$\dfrac{\dfrac{x \vee y \qquad \dfrac{x \vee \bar{y} \vee z \qquad \bar{z}}{x \vee \bar{y}}}{x} \qquad \dfrac{\bar{x} \vee z \qquad \bar{z}}{\bar{x}}}{\square}$$

$x \vee y$	$x \vee \bar{y} \vee z$	$\bar{z}$	$\bar{x} \vee z$	$x \vee \bar{y}$	x	$\bar{x}$	$\square$

Remarks

Improvements of the "interface" checker:

- all the clauses are not alive at the same time
- efficient representation of clauses

"Small" checkers:

resolution: efficient:

- computation of resolution chains
- representation of clauses

theories: two approches:

- detailed proof witnesses
- decision procedure in Coq

Benchmarks coming from the SAT- and SMT-comp

ZChaff on 151 benchmarks from SAT Race'06 and '08

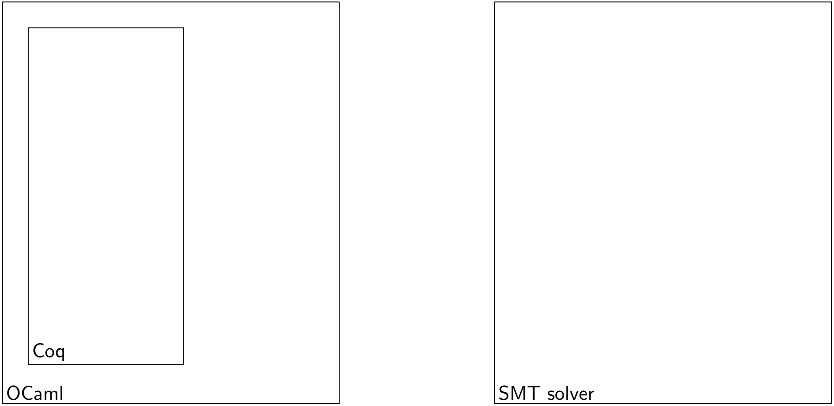
Solved ZChaff			Coq checker			Isabelle/HOL checker		
#	%	Time	#	%	Time	#	%	Time
75	49.7	64.3	70	46.4	22.3	57	37.7	101.

VeriT (SAT+congruence closure) on 4019 benchmarks from SMT-LIB

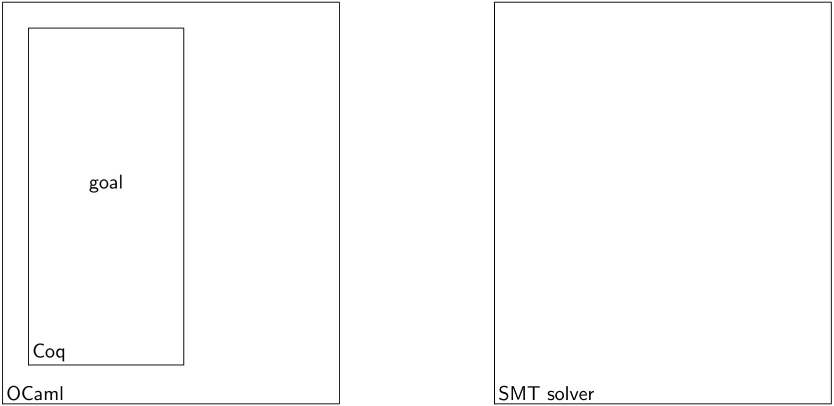
Solved VeriT			Coq checker		
#	%	Time	#	%	Time
3897	97.0	5.075	3871	96.3	1.050

Outline

Idea

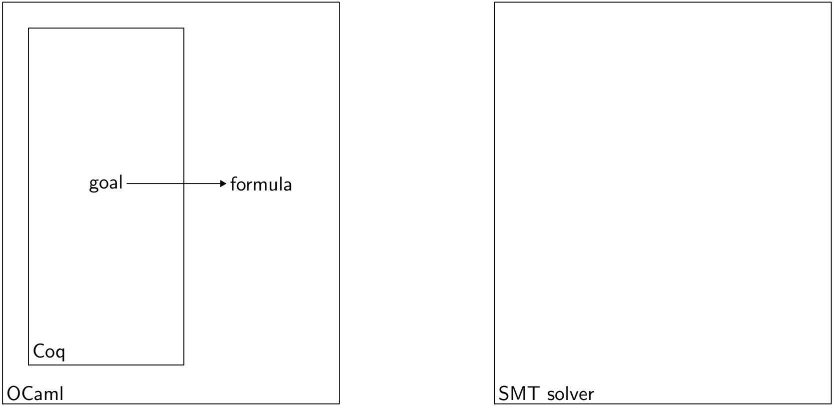


Idea



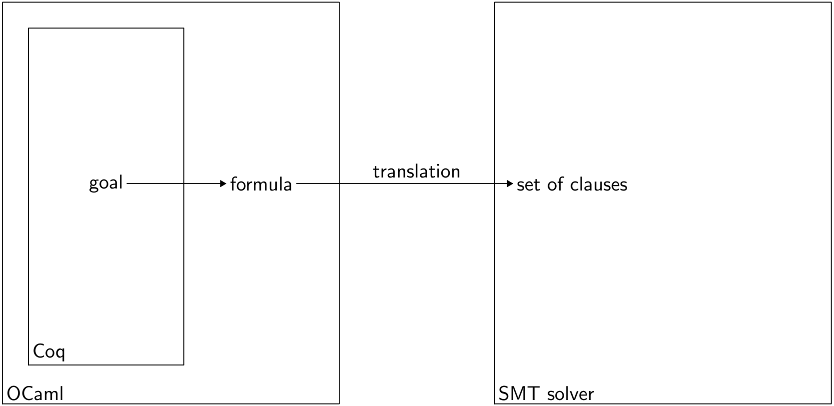
$\forall \vec{x}, F$ is true

Idea



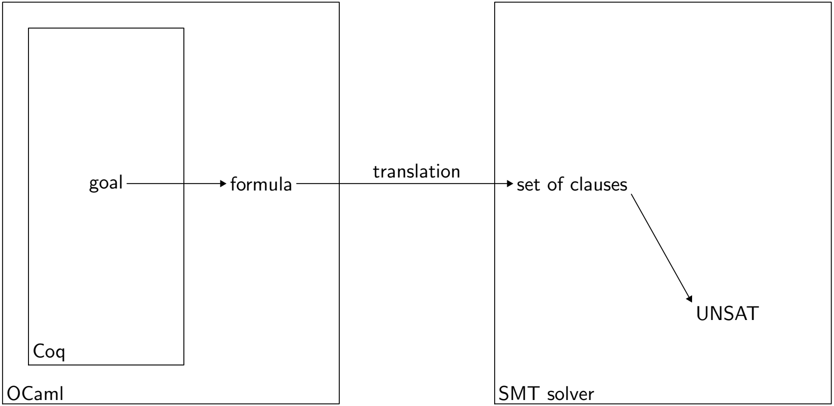
$\forall \vec{x}, F$ is true $\Leftrightarrow \exists \vec{x}, \neg F$ is false

Idea



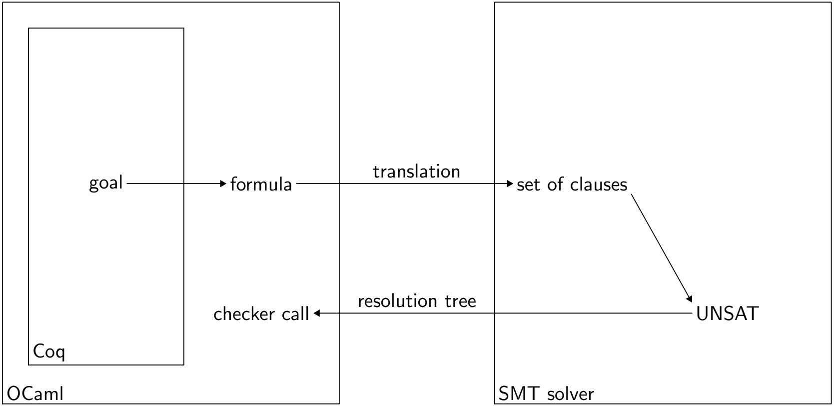
$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Idea



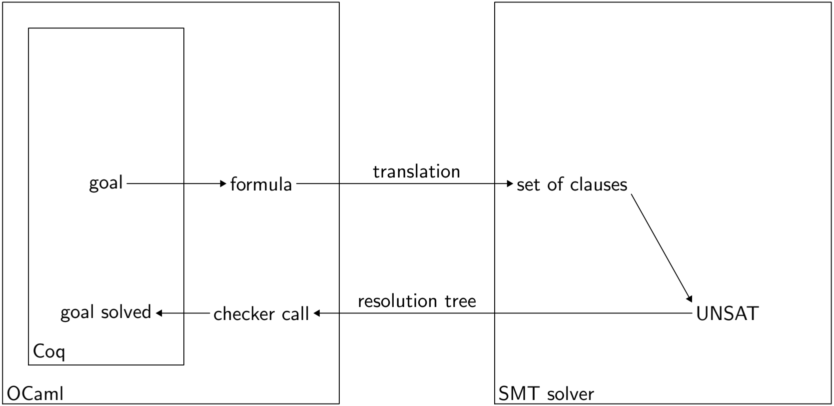
$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Idea



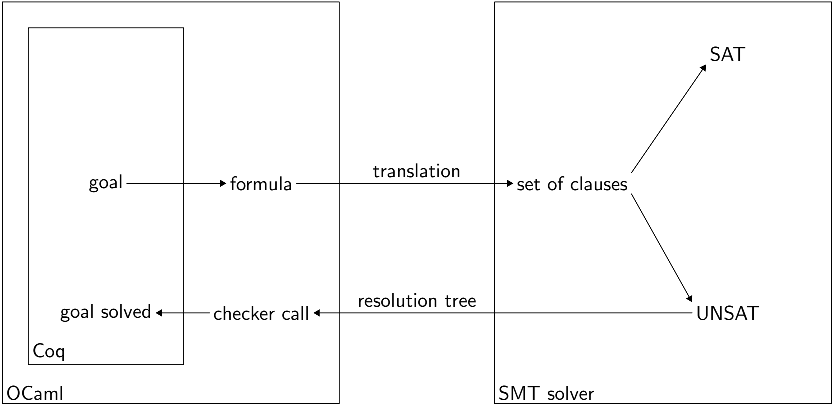
$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Idea



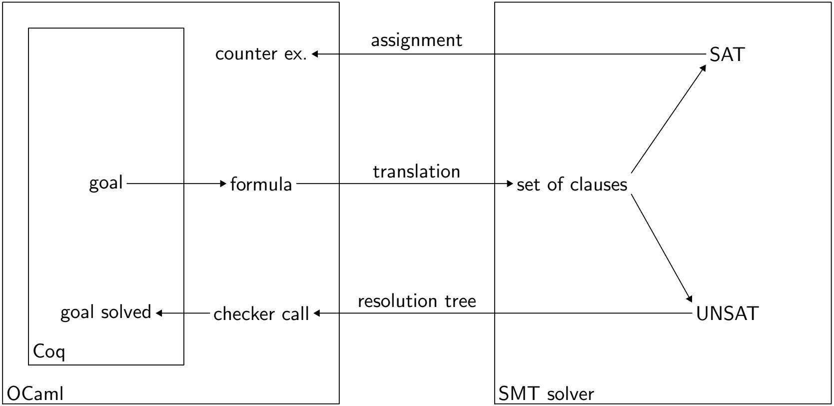
$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Idea



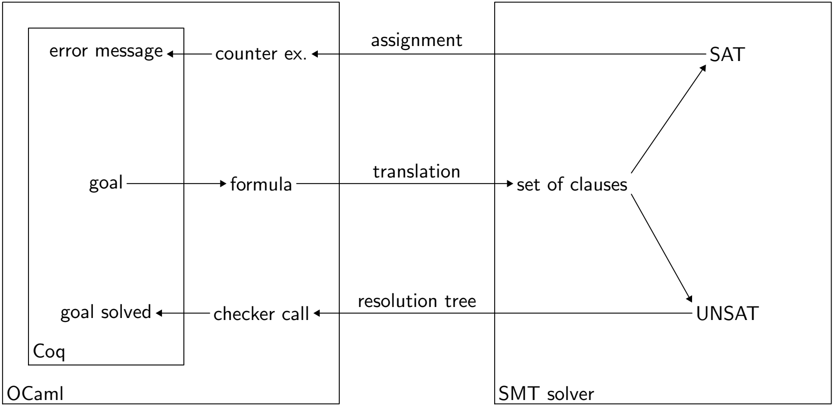
$\forall \vec{x}, F$ is true $\Leftrightarrow \exists \vec{x}, \neg F$ is false $\Leftrightarrow \neg F$ is unsatisfiable

Idea



$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Idea



$$\forall \vec{x}, F \text{ is true} \Leftrightarrow \exists \vec{x}, \neg F \text{ is false} \Leftrightarrow \neg F \text{ is unsatisfiable}$$

Outline

Related works

Proof witness verification:

- S. Böhme and T. Weber: verification in HOL and Isabelle/HOL
- F. Besson et al.: combination of theories in Coq
- P. Fontaine et al.: Harvey in Isabelle/HOL

SMT solvers certification:

- S. Lescuyer et al.: embedding Alt-Ergo in Coq

Conclusion and perspectives

Conclusion:

- efficient *a posteriori* verification of SMT solvers
- new decision procedure in Coq

Perspectives:

- encoding of more expressive Coq terms
- quantifiers
- new theories

Thanks

Thank you for listening! Any questions?

- Website: http://www.lix.polytechnique.fr/~keller/Recherche/smtcoq.html
- Demo: yes!