

Coq tactic

Conclusion

Cooperation between SAT, SMT provers and Coq

Michaël Armand <u>Chantal Keller</u>

Germain Faure Benjamin Grégoire Laurent Théry Benjamin Werner

INRIA – École Polytechnique

April, 2nd 2011





Tools

 $\underset{\bigcirc}{\text{Coq tactic}}$

Conclusion

Introduction



Tools

 $\underset{\bigcirc}{\text{Coq tactic}}$

Conclusion

Introduction



To ol s 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
Introduction			



Tools 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
Introduction			



Tools 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
Introduction			



To ol s 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
Introduction			



To ol s 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0

Outline



2 Coq checker

3 Coq tactic



Cooperation between SAT, SMT provers and Coq



Proof producing SAT solvers

Decide propositional satisfiability of sets of clauses:

 $\blacksquare x \lor y \qquad x \lor \overline{y} \lor z \qquad \overline{x} \lor z \qquad \overline{z}$

Proof witness:

- \blacksquare If satisfiable: assignment of the variables to op or op
- If unsatisfiable: proof by resolution of the empty clause

Resolution rule:

$$\frac{x \lor C \quad \bar{x} \lor D}{C \lor D}$$

Tools ⊙● ○○○	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
00 SAT solvers			
Examples			
Satisfiability of:	$x \lor y$ $x \lor \overline{y} \lor$	$\bar{z} \bar{x} \lor z$	
	$\{x \mapsto \top, y \mapsto \bot, z \in$	$\mapsto \top \}$	
Unsatisfiability	of: $x \lor y$ $x \lor \overline{y} \lor$	\overline{z} $\overline{x} \lor \overline{z}$ \overline{z}	
	$\underbrace{x \lor \bar{y} \lor z \qquad \bar{z}}_{}$		
$x \vee$	$y x \lor \overline{y}$	$\overline{\overline{x} \lor z} \overline{\overline{z}}$	
	X	x	

Tools O● OOO	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0
00 SAT solvers			
Examples			
Satisfiability of:	$x \lor y$	$x \lor \overline{y} \lor z \qquad \overline{x} \lor z$	
	$\{x\mapsto op, y\mapsto$	$\rightarrow \perp, z \mapsto \top \}$	
Unsatisfiability	of: $x \lor y$	$x \lor \bar{y} \lor z \qquad \bar{x} \lor z$	Ī
	$x \vee \overline{y} \vee z$	Ī	
$x \vee$	$y x \vee \overline{y}$	$\overline{x} \lor z$ \overline{z}	-
	X	x	_
	E		

 Tools
 Coq checker
 Coq tactic
 Conclusion

 000
 0000
 0000
 0000

 000
 0000
 0000
 0000

 SMT solvers
 0000
 0000

Proof producing SMT solvers

Atoms are now formulas of some theories:

- congruence closure
- linear arithmetic
- **.**..

•
$$f(x) \neq f(y)$$
 $f(x) = f(f(z))$ $x = y$

Proof witness:

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

Tools	Coq checker	Coq tactic	Conclusion
00			
SMT solvers			

Examples

Satisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) $\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$ Unsatisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) x = y $x \neq y \lor f(x) = f(y)$ x = yf(x) = f(y) $f(x) \neq f(y)$ \square

Tools	Coq checker	Coq tactic	Conclusion
00			
SMT solvers			

Examples

Satisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) $\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$ Unsatisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) x = y $x \neq y \lor f(x) = f(y)$ x = yf(x) = f(y) $f(x) \neq f(y)$ \square

Tools	Coq checker	Coq tactic	Conclusion
00			
SMT solvers			

Examples

Satisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) $\{x \mapsto f(a), y \mapsto a, z \mapsto a\}$ Unsatisfiability of: $f(x) \neq f(y)$ f(x) = f(f(z)) x = y $x \neq y \lor f(x) = f(y)$ x = yf(x) = f(y) $f(x) \neq f(y)$ \square

Tools ⊙ ⊙O● ⊙O	Coq checker 0000 0	Coq tactic ⊙	Conclusion 00 0
SMT solvers			

Such tools

SAT solvers MiniSat zChaff

SMT solvers







Tools	Cog checker	Coq tactic	Conclusion
00 000 • 0			
Coa			

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)

Tools	Cog checker	Cog tactic	Conclusion
00 000 ● 0			
Coa			

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

Tools	Cog checker	Coq tactic	Conclusion
00 000 00			





2 Coq checker

3 Coq tactic



Cooperation between SAT, SMT provers and Coq

Tools	Cog checker	Coq tactic	Conclusion
00	0000		
000			
Main idea			

SAT and UNSAT

Main idea:

- \blacksquare 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

Tools	Coq checker	Coq tactic	Conclusion
	0000		
Main idea			

SAT and UNSAT

Main idea:

- \blacksquare 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

Tools	Coq checker	Coq tactic	Conclusion
	0000		
Main idea			

SAT and UNSAT

Main idea:

- \blacksquare 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

Tools	Cog checker	Coq tactic	Conclusion
	0000		
000			
Main idea			

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



Coq tactic O Conclusion



Fools DO DOO DOO	Coq checker 00●0 0	Coq tactic ○	Conclusion 00 0
Alle tales			



8 different clauses:

Tools	Cog checker	Coq tactic	
00 000 00	0000 0		



 Tools
 Cog checker
 Cog tactic
 Conclusion

 0000
 0000
 0000
 0000

 Main idea
 0000
 0000



ools		

Coq tactic 0 Conclusion



$x \lor y$	$x \lor \overline{y} \lor z$	Ī	$\bar{x} \lor z$				
------------	------------------------------	---	------------------	--	--	--	--

ools			



$$x \lor y$$
 $x \lor \overline{y} \lor z$
 \overline{z}
 $\overline{x} \lor z$
 \uparrow
 \uparrow

To ol s			

Coq checker ○○●○ Coq tactic 0 Conclusion

$$\underline{x \lor y} \quad \frac{x \lor \overline{y} \lor z \quad \overline{z}}{x \lor \overline{y}} \quad \overline{x \lor z \quad \overline{z}}$$

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

Tools	Cog checker	Coq tactic	
00 000 00	0000 0		

$$\frac{x \lor y}{x} \frac{\frac{x \lor \overline{y} \lor z}{x \lor \overline{y}}}{x} \frac{\overline{z}}{\overline{x}} \frac{\overline{x} \lor z}{\overline{z}}}{\overline{x}}$$

$x \lor y$	$x \lor \overline{y} \lor z$	Ī	$\bar{x} \lor z$	$x \lor \bar{y}$			
------------	------------------------------	---	------------------	------------------	--	--	--

Tools	Cog checker	Coq tactic	
00 000 00	00 ●0 ○		

$$\frac{x \lor y}{x} \frac{\begin{array}{c} x \lor \overline{y} \lor z & \overline{z} \\ x \lor \overline{y} \end{array}}{x} \frac{\overline{x} \lor z & \overline{z}}{x}$$

$$x \lor y$$
 $x \lor \overline{y} \lor z$
 \overline{z}
 $\overline{x} \lor z$
 $x \lor \overline{y}$
 \uparrow
 \uparrow

Tools Coq checker 00 00●0 000 0		Coq tactic O	Conc 00 0
Main idea			

$$\frac{x \lor y}{x} \frac{\begin{array}{c} x \lor \overline{y} \lor z & \overline{z} \\ x \lor \overline{y} \end{array}}{x} \frac{\overline{x} \lor z & \overline{z}}{x}$$

$x \lor y$	$x \lor \overline{y} \lor z$	Ī	$\bar{x} \lor z$	$x \vee \overline{y}$	х	
					Ť	

usion

Tools	Cog checker	Coq tactic
	0000	

Conclusion

The "interface" checker by Example

	$x \vee \overline{y} \vee z$	Ī		
$x \lor y$	$x \vee \bar{y}$		$\bar{x} \lor z$	Ī
	X		\overline{x}	

$x \lor y$	$x \vee \bar{y} \vee z$	Ī	$\bar{x} \lor z$	$x \vee \overline{y}$	х	

Tools	Cog checker	Coq t
	0000	

Conclusion

The "interface" checker by Example

$$\frac{x \lor y}{x} \frac{x \lor \overline{y} \lor z \quad \overline{z}}{x \lor \overline{y}}}{x} \frac{\overline{x} \lor z \quad \overline{z}}{\overline{x}}$$

Tools	Cog checker	Coq tactic
	0000	

Conclusion

The "interface" checker by Example



$x \lor y$	$x \lor \overline{y} \lor z$	Ī	$\bar{x} \lor z$	$x \lor \bar{y}$	X	x	
						↑	

Fool s			

Coq tactic ○



Fool s			

Coq checker

Coq tactic 0 Conclusion



Tools			

Coq checker

Coq tactic ○ Conclusion





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

Coq tactic ○ Conclusion

Benchmarks coming from the SAT- and SMT-comp

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

Sc	Solved ZChaff			Coq checker		Isab	elle/H	OL 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			Co	oq chec	ker
#	%	Time	#	%	Time
3897	97.0	5.075	3871	96.3	1.050

Tools 00 000 00	Coq checker 0000 0	Coq tactic ○	Conclusion 00 0





2 Coq checker

3 Coq tactic



Tools	Cog checker	Coq tactic	Conclusion
00 000 00		•	
Main idea			



Tools	Cog checker	Coq tactic	Conclusion
00 000 00		•	
Main idea			



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

Tools	Coq checker	Coq tactic	Conclusion
00 000 00		•	
Main idea			



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

Tools 00 000 00	Cog checker 0000 0	Coq tactic ●	Conclusion 00 0
Main idea			







Tools	Coq checker	Coq tactic	Conclusion
00 000 00		•	
Main idea			



To ol s 00 000 00	Coq checker 0000 0	Coq tactic ●	Conclusion 00 0
Main idea			







Tools	Cog checker	Coq tactic	Conclusion
		•	
000			
Main idea			



Tools	Coq checker	Coq tactic	Conclusion
		•	
000			
00			
Main idea			



Tools 00 000 00	Coq checker 0000 0	Coq tactic O	Conclusion 00 0





2 Coq checker

3 Coq tactic



Tools	Coq checker	Coq tactic	Conclusion
00			• • • •
Related works, conclus	sion and perspectives		

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

Tools 00 000	Соq checker 0000 0	Coq tactic 0	Conclusion ⊙● ⊙
Related works con	lucion and perspectives		

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

Tools	Coq checker	Coq tactic	Conclusion
00			00
000			•
Thanks			

Thanks

Thank you for listening! Any questions?

Website: http://www.lix.polytechnique.fr/~keller/Recherche/smtcoq.html

Demo: yes!