

SAT Modulo Theories and Lazy Clause Generation

Ian Gent

University of St Andrews

with enormous thanks to

Neil Moore, Peter Stuckey

Topics in this Series

- Why SAT & Constraints?
- SAT basics
- Constraints basics
- Encodings between SAT and Constraints
- Watched Literals in SAT and Constraints
- Learning in SAT and Constraints
- Lazy Clause Generation + SAT Modulo Theories

SMT & LCG

- SAT Modulo Theories
- Lazy Clause Generation
- Two areas where constraints/SAT most tightly integrated

SAT Modulo Theories

- Extremely important & trendy area
- Will dismiss it in a few slides
 - because ...
- Integration with Constraints not so tight ... and
- Nieuwenhuis & Stuckey agree they are converging together ... and
- I am running out of time ... and
- I know very little about it ... and
- I like “Lazy Talk Generation”

SMT

- Idea is simple
- SAT is very good at search on literals
- We may have theory reasoners good at their thing
- Combine the two
 - Search using SAT
 - Reason using theory reasoner

SMT example

- Simple example based on software checking
 - solution means type error
- SAT literals are
 - $x > 0$
 - $y > 0$
 - $x > y$
 - $y - x \leq 0$
 - $42 \leq 0$ [= false]

$$x > 0$$

$$\wedge y > 0$$

$$\wedge (x > y \rightarrow y - x \leq 0)$$

$$\wedge (x \leq y \rightarrow 42 \leq 0)$$

SMT example

- Simple example based on software checking
 - solution means type error
- SAT literals are
 - searched on as if simple literals
 - i.e. SAT solvers knows nothing about inequalities
 - solution found by assigning literals true / false

$$x > 0$$

$$\wedge y > 0$$

$$\wedge (x > y \rightarrow y - x \leq 0)$$

$$\wedge (x \leq y \rightarrow 42 \leq 0)$$

SMT example

- SAT might search to get...
 - $x > 0$ **T** $x > 0$
 - $y > 0$ **T** $\wedge y > 0$
 - $x > y$ **F** $\wedge (x > y \rightarrow y - x \leq 0)$
 - $y - x \leq 0$ **T** $\wedge (x \leq y \rightarrow 42 \leq 0)$
 - $42 \leq 0$ **T**

SMT example

- SAT literals are
 - searched on as if simple literals
 - i.e. SAT solvers knows nothing about inequalities
 - solution found by assigning literals true / false
- SAT solution given to theory reasoner
 - which either
 - accepts it
 - or fails and gives explanation
- Explanations are clauses and treated as in SAT solving

$$x > 0$$

$$\wedge y > 0$$

$$\wedge (x > y \rightarrow y - x \leq 0)$$

$$\wedge (x \leq y \rightarrow 42 \leq 0)$$

SMT example

- SAT might search to get...

$$x > 0$$

- ...

- $x > y$ **F**

$$\wedge y > 0$$

- $42 \leq 0$ **T**

$$\wedge (x > y \rightarrow y - x \leq 0)$$

- Fail!

$$\wedge (x \leq y \rightarrow 42 \leq 0)$$

- Learnt clause might be

- $x > y$

SMT example

- SAT literals are
 - $x > 0$ **T**
 - $y > 0$ **T**
 - $x > y$ **T**
 - $y - x \leq 0$ **T**
 - $42 \leq 0$ **F**
- Theory reasoner succeeds!
 - sets e.g. $x=9, y=10$
 - so software bugged

$$x > 0$$

$$\wedge y > 0$$

$$\wedge (x > y \rightarrow y - x \leq 0)$$

$$\wedge (x \leq y \rightarrow 42 \leq 0)$$

SMT Solving

- Search in SAT
- Theory Decisions
- Theory Propagation
 - not seen in toy example
 - tell theory about every decision
 - let it propagate (if it can)
 - and communicate results by explanation
- Learning and Theory Explanations

SMT vs Constraints

- SMT has broader reach than constraints
 - Can use any theory we like
 - If we can code it as a theory
 - Many theories encoded into SMT
 - Theories need not even be decidable
 - so more general than Constraints
- SMT has less coverage in constraints
 - Constraint type theories not well developed
 - E.g. Alldiff theory only recently produced (2010??)
 - Not clear that propagation as effective

Lazy Clause Generation

- Alternative track of research
 - starting with
 - Ohrimenko, Stuckey, and Codish, 2007
- But as optimisations made on both sides
 - SMT and LCG converging
- So SMT more general
- But LCG much better at constraint problems

Lazy Talk Generation

- Dear Peter ...

"I wonder if there's a lazy solution for me, to steal some of your slides on lazy clause generation? This would save me lots of time and also let me present it accurately instead of wrongly! It won't go without saying I'd give you full credit for the slides."

- Slides available from

<http://ww2.cs.mu.oz.au/~pjs/637/lec/>