@Piccola Radge

Propositional logic + theories **SMT solving** (SAT-modulo-theories solving)

Satisfiability equivalent transformation

transformation

EAGER SMT solving — theories

LAZY SMT solving

Equality logic and uninterpreted functions (EQ+UF)

Finite-precision bit-vector arithmetic

FULL lazy SMT solving — full solution

LESS lazy SMT solving — partial solution

Equality logic „φ^UF" with uninterpreted functions

bit blasting (or „flattening")

*Sparse method

Ackermann's reduction

SAT solver + Theory solver

→ Requirements:
- Incrementality
- (preferably minimal) infeasible subset
- Backtracking

Equality logic „φ^EQ" (without uninterpreted functions)

φ^EQ in NNF
Equality Graph (E-Graph)

also called „real algebra"

Non-linear real arithmetic

Linear real arithmetic (or also „linear real algebra")

Quantifier-free NONlinear arithmetic (QFNRA)

Cylindrical algebraic decomposition (CAD) COMPLETE

Quantifier-free linear real arithmetic (QFLRA)

Variable elimination

SATcheck

Satisfiability Checking

Interval Constraint Propagation (ICP) INCOMPLETE

Gauß method (based on equations)

Fourier Motzkin

Virtual substitution (VS) INCOMPLETE

Subtropical satisfiability INCOMPLETE

Simplex

Integer linear systems

integer solutions

SATcheck

SAT solving

Branch and bound algorithm

SAT solving algorithms

Enumeration algorithm

DPLL algorithm

DPLL+CDCL (conflict driven clause learning)

Decision heuristics

+ Boolean Constraint Propagation (BCP)

+ conflict resolution

conflict-driven backtracking

static ---- + ---- dynamic

Jeroslow-Wang method

Dynamic Largest Individual Sum (DLIS)

watched literals

quasi-static

Variable State Independent Decaying Sum (VSIDS)

*The sparse method is one of 2 algorithms to transform equality logic to propositional logic. The difference: The sparse method is based on polar E-graphs, whereas the 2nd algorithm mentioned in lecture 5 is based on non-polar E-graphs

Lecture topics:

1. ● Propositional Logic (VL2 – VL4)
3. ● FO logic (VL7)

---

2. ● **SAT solving (VL5-VL6)**

~~4. (● Decidability (VL8)) → not relevant for exam!~~

5. ● Eager SMT solving: Equality logic and uninterpreted functions (VL9)
6. ● Eager SMT solving for finite-precision bit vectors arithmetic (VL10)
7. ● Lazy SMT solving (VL12)
8. ● (Full/less) lazy SMT solving for equality logic (VL13)

SMT solving

9. ● Gauß and Fourier-Motzkin variable elimination for linear real arithmetic (VL14)
10. ● The Simplex Algorithm (VL15)
11. ● Simplex in SMT solving (VL16)
12. ● Branch and bound (VL17)

Satisfiability Checking of linear real arithmetic and linear integer arithmetic

13. ● Interval Constraint Propagation (ICP) (VL19 – VL20)
14. ● Subtropical satisfiability (VL21)
15. ● The decomposition idea for solving real arithmetic problems (VL22)
16. ● Virtual substitution (VS) (VL23 – VL24)
17. ● Cylindrical algebraic decomposition (CAD) (VL25 – VL26)

Satisfiability checking of QFNRA