EECS 219C: Computer-Aided Verification

Satisfiability Modulo Theories

Theory Solvers, Combination of Theories

Sanjit A. Seshia
EECS, UC Berkeley

Topics for Today

- Examples of Theory Solvers
  - Equality and Uninterpreted Functions
  - Difference Logic
  - Arrays
- Combination of Theories
  - The Nelson-Oppen Framework

Conjunctions Only

- Focus: solving conjunctions of constraints
- Expect: no case-splitting
  - But: Sometimes need to perform case-splitting even if there are no disjunctions (ORs) to start with!

Theory of Equality

- The theory of equality is all known as the free/empty theory.
- The theory does not restrict the possible values of symbols in any way. For this reason, it is sometimes called the theory of equality with uninterpreted functions (EUF).
- The satisfiability problem for conjunctions of literals in this theory is decidable in polynomial time using *congruence closure*.

- Example:

$$g(g(g(x))) = x$$
$$\wedge\ g(g(g(g(g(x))))) = x$$
$$\wedge\ g(x) \neq x$$

S. A. Seshia 5

Properties of Congruence Closure

- Does it always terminate? Why?

S. A. Seshia 6

Difference Logic

- Recall that a difference constraint is a linear constraint of the form

  $x_i - x_j \leq c$ or $x_i \leq c$ or $-x_i \leq c$
- Consider a system of such constraints. How do we solve this?
- We can turn this into a shortest path problem!

Constraint Graph

- Add a node for every variable x_i
- Given the constraint $x_i - x_j \leq c$, add an edge from node i to node j labeled with c
- Questions:
  - What about $\pm x_i \leq c$?
  - What if we have two constraints of the form $x_i - x_j \leq c_1$ and $x_i - x_j \leq c_2$?

Example: SAT or UNSAT?

$x_1 \geq x_2$

$x_3 \leq 0$

$x_2 + 3 \geq x_1$

$x_1 + 1 \leq x_3$

$x_2 + 1 \geq 0$

$x_4 + 2 \geq 0$

$x_4 \leq x_2 - 2$

Algorithm

- Theorem: A set of difference constraints is satisfiable iff there is no negative-weight cycle in the constraint graph
- Proof:

(a) soundness: if there is a neg-wt cycle, then the constraints are really unsat:

Draw out a negative weight cycle – what constraint is implied by it?

Algorithm

- Theorem: A set of difference constraints is satisfiable iff there is no negative-weight cycle in the constraint graph
- Proof:

(b) completeness: if there is no neg-wt cycle, then the constraints are satisfiable:

Intuition: (i) we can compute the shortest paths between any x_i and x_0; (ii) the length of the shortest path from x_0 to x_i gives the satisfying assignment for x_i

Complexity

- What is the asymptotic running time of the shortest-path based algorithm for difference logic?

Theory of Arrays

- Two main axioms: For all A, i, j, d
  – select(store(A,i,d), i) = d
  – select(store(A,i,d), j) = select(A,j), if i ≠ j
- Decision procedure operates by performing case-splits
- Example:

```
int a[10];
int fun3(int i) {
   int j;
   for(j=0; j<10; j++) a[j] = j;
   assert(a[i] <= 5);
}
```


Theory of Arrays

- Two main axioms: For all A, i, j, d
  – select(store(A,i,d), i) = d
  – select(store(A,i,d), j) = select(A,j), if i ≠ j
- Decision procedure operates by performing case-splits
- Example:

$a[0] = 0 \wedge a[1] = 1 \wedge a[2] = 2 \wedge \ldots a[10] = 10 \wedge a[i] > 5$

Nelson-Oppen Framework

- Often, the SMT problem is not expressible in a single theory
- Therefore we need to combine decision procedures for different theories to work on the combined theory
- The main approach for this purpose is the one proposed by Nelson and Oppen in '79