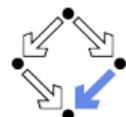


# SMT SOLVING: COMBINING DECISION PROCEDURES

Course “Computational Logic”



Wolfgang Schreiner  
Research Institute for Symbolic Computation (RISC)  
[Wolfgang.Schreiner@risc.jku.at](mailto:Wolfgang.Schreiner@risc.jku.at)



## Lemmas on Demand

How to decide  $T \models F$  for unquantified formula  $F$  and decidable theory  $T$ ?

- **So far:** convert  $F$  into a disjunctive normal form  $C_1 \vee \dots \vee C_n$ .
  - $F$  is  $T$ -satisfiable if and only if some  $C_i$  is  $T$ -satisfiable.
  - Problem: the number  $n$  of clauses may be exponential in the size of  $F$ .
- **Better:** combine the decision procedure for  $T$  with a *SAT solver*.
  - The SAT solver is applied to the **propositional skeleton**  $\overline{F}$ .
    - Every atomic formula  $A$  in  $F$  is abstracted to a propositional variable  $\overline{A}$ .
    - If  $\overline{F}$  is unsatisfiable,  $F$  is unsatisfiable and we are done.
    - Otherwise, the SAT solver produces a satisfying assignment represented by a conjunction  $\overline{L_1} \wedge \dots \wedge \overline{L_m}$  of propositional literals.
  - The decision procedure is applied to the  $T$ -formula  $L_1 \wedge \dots \wedge L_m$ .
    - Propositional variable  $\overline{L_i}$  is expanded into the atomic formula  $L_i$  it represents.
    - If the formula is satisfiable,  $F$  is satisfiable and we are done.
    - Otherwise, the decision procedure determines a minimal unsatisfiable subformula  $C$  of  $L_1 \wedge \dots \wedge L_m$  and we repeat the process with  $F \wedge \neg C$ .

Each formula  $\neg C$  produced represents a “lemma” deduced from  $F$ .

# Example

$E$ -satisfiability of  $F : \Leftrightarrow x = y \wedge ((y = z \wedge x \neq z) \vee x = z)$ .

- First iteration:

- Propositional skeleton:  $a \wedge ((b \wedge \neg c) \vee c)$
- Satisfying assignment:  $a \wedge b \wedge \neg c$
- Unsatisfiable concretization:  $x = y \wedge y = z \wedge x \neq z$
- Strengthened formula:  $F \wedge \neg(x = y \wedge y = z \wedge x \neq z)$

- Second iteration:

- Propositional Skeleton:  $a \wedge ((b \wedge \neg c) \vee c) \wedge \neg(a \wedge b \wedge \neg c)$
- Satisfying assignment:  $a \wedge b \wedge c$
- Satisfiable concretization:  $x = y \wedge y = z \wedge x = z$

Formula  $F$  is  $E$ -satisfiable.

# Algorithm

```
function SMT( $F$ )                                ▷ decides  $T$ -satisfiability of formula  $F$ 
   $\overline{F}$  := ABSTRACT( $F$ )
  loop
    ( $sat, \overline{Ls}$ ) := SAT( $\overline{F}$ )           ▷ decides satisfiability of propositional skeleton of  $F$ 
    if  $\neg sat$  return false
     $Ls$  := CONCRETIZE( $\overline{Ls}$ )
    ( $sat, C$ ) := SMT( $Ls$ )                     ▷ decides  $T$ -satisfiability of literal set  $Ls$ 
    if  $sat$  return true
     $\overline{F}$  :=  $\overline{F} \wedge$  ABSTRACT( $\neg C$ )
  end loop
end function
```

This basic approach can be further optimized, e.g., by integrating the interaction with the decision procedure directly into a DPLL-based SAT solver (“lazy encoding”).

# Combining Decision Procedures

How to decide a conjunction of atomic formulas with operations from different decidable theories such as **LRA** and **EUF**?

$$(y \geq z) \wedge (x - z \geq y) \wedge (z \geq 0) \wedge (f(f(x) - f(y)) \neq f(z))$$

- **Theory combination problem:** decide  $T_1 \cup T_2 \models F$  for formula  $F$  and theories  $T_1, T_2$ .
  - Problem: even if  $T_1$  and  $T_2$  are decidable,  $T_1 \cup T_2$  may be undecidable.
- **Definition:** a theory  $T$  is **stably infinite**, if for every quantifier-free formula  $F$  that is  $T$ -satisfiable, there exists an infinite domain that satisfies  $F$ .
  - Theories *LRA* and *EUF* are stably infinite.
  - The theory  $\{x = a \vee x = b\}$  with constants  $a, b$  is not stably infinite (why?).
- **Theorem:** let  $T_1$  and  $T_2$  be stably infinite theories for which the quantifier-free fragment is decidable and that have no common constants, functions, or predicates (except for “ $=$ ”). Then the quantifier-free fragment of  $T_1 \cup T_2$  is decidable.

For stably infinite theories, the theory combination problem is indeed solvable.

# Formula Purification

Before proceeding, let us tidy the formula a bit.

- **Purification:** ensure that every atom is from only one theory.
  - Repeatedly replace in the formula each “alien” subexpression  $E$  by a fresh variable  $v_E$  and add the constraint  $v_E = E$ .
  - The transformation preserves the satisfiability of the formula.
- **Example:**  $(f(x, 0) \geq z) \wedge (f(y, 0) \leq z) \wedge (x \geq y) \wedge (y \leq x) \wedge (z - f(x, 0) \geq 1)$ .

$$(v_1 \geq z) \wedge (v_2 \leq z) \wedge (x \geq y) \wedge (y \leq x) \wedge (z - v_1 \geq 1) \wedge \\ v_1 = f(x, v_3) \wedge v_2 = f(y, v_3) \wedge v_3 = 0$$

A preparatory step for theory combination.

# The Nelson-Oppen Method (for Convex Theories)

Greg Nelson and Derek C. Oppen (1979).

```
function NELSONOPPEN( $F$ )
   $F_1, \dots, F_n := \text{PURIFY}(F)$                                 ▷ decides  $T_1 \cup \dots \cup T_n$ -satisfiability of literal conjunction  $F$ 
  loop
    if  $\exists i. \neg \text{SMT}_i(F_i)$  return false                      ▷ for convex theories  $T_1, \dots, T_n$ 
    if  $\neg \exists x, y, j. \text{INFERRED}_j(x, y)$  return true
    choose  $x, y, j$  with  $\text{INFERRED}_j(x, y)$     ▷ infer variable equality  $x = y$  not present in theory  $T_j$ 
     $F_j := F_j \cup \{x = y\}$                                          ▷ propagate inferred variable equality to  $T_j$ 
  end loop
end function
```

$\text{INFERRED}_j(x, y) : \Leftrightarrow \exists i. (\text{SHARED}(F_i, F_j, \{x, y\})) \wedge \text{INFER}_i(F_i, (x = y)) \wedge \neg \text{INFER}_j(F_j, (x = y))$

- $\text{SHARED}(F_i, F_j, \{x, y\})$ : variables  $x, y$  are shared by formulas  $F_i$  and  $F_j$ .
- $\text{INFER}_i(F_i, (x = y))$ : variable equality  $(x = y)$  can be inferred from  $F_i$  in theory  $T_i$ .
  - $F_i \Rightarrow x = y$  is  $T_i$ -valid ( $F_i \wedge \neg(x = y)$  is  $T_i$ -unsatisfiable).

The iterative propagation of inferred variable equalities between theories.

# Example

$$(f(x, 0) \geq z) \wedge (f(y, 0) \leq z) \wedge (x \geq y) \wedge (y \geq x) \wedge (z - f(x, 0) \geq 1)$$

- Purified formula:

$$\begin{aligned} & (v_1 \geq z) \wedge (v_2 \leq z) \wedge (x \geq y) \wedge (y \geq x) \wedge (z - v_1 \geq 1) \wedge \\ & v_1 = f(x, v_3) \wedge v_2 = f(y, v_3) \wedge v_3 = 0 \end{aligned}$$

- Equality propagation:

$$\begin{array}{c} \begin{array}{cc} \hline F_1(LRA) & F_2(EUF) \\ \hline v_1 \geq z & v_1 = f(x, v_3) \\ v_2 \leq z & v_2 = f(y, v_3) \\ x \geq y & \\ y \geq x & \\ z - v_1 \geq 1 & \\ v_3 = 0 & \hline \end{array} \\ \begin{array}{ccc} \frac{x = y}{v_1 = v_2} & \rightarrow & x = y \\ & \leftarrow & \frac{v_1 = v_2}{v_1 = z} \\ & & \hline \end{array} \\ \text{unsat} \end{array}$$

# Example

$$(y \geq x) \wedge (x - z \geq y) \wedge (z \geq 0) \wedge (f(f(x) - f(y)) \neq f(z))$$

- Purified formula:

$$(y \geq x) \wedge (x - z \geq y) \wedge (z \geq 0) \wedge (f(v_1) \neq f(z)) \wedge \\ v_1 = v_2 - v_3 \wedge v_2 = f(x) \wedge v_3 = f(y)$$

- Equality propagation:

$$\begin{array}{c} \begin{array}{cc} \hline F_1(LRA) & F_2(EUF) \\ \hline y \geq x & f(v_1) \neq f(z) \\ x - z \geq y & v_2 = f(x) \\ z \geq 0 & v_3 = f(y) \\ \hline v_1 = v_2 - v_3 & \\ \hline z = 0 & \\ \hline x = y & \rightarrow \quad x = y \\ v_2 = v_3 & \leftarrow \quad \underline{v_2 = v_3} \\ v_1 = 0 & \\ \hline \underline{v_1 = z} & \rightarrow \quad v_1 = z \\ & \text{unsat} \end{array} \end{array}$$

# Convex Theories

- **Definition:** Theory  $T$  is **convex**, if for every formula  $F := L_1 \wedge \dots \wedge L_m$  with literals  $L_1, \dots, L_m$  the following holds (for variables  $x_1, \dots, x_n$  and  $y_1, \dots, y_n$ ):
  - If  $T \models F \Rightarrow x_1 = y_1 \vee \dots \vee x_n = y_n$ , then  $T \models (F \Rightarrow x_i = y_i)$  for some  $i \in \{1, \dots, n\}$ .
    - If  $F$  implies in  $T$  a disjunction of equalities, it already implies one of these equalities.
    - Thus  $F$  cannot express “real” disjunctions and it suffices to infer plain equalities.
- **Examples:**
  - **LRA is convex:** a “real” disjunction corresponds to a finite set of  $n \geq 2$  geometric points; however, by a conjunction of linear inequalities (which represents an intersection of half-planes), we can only define a point set that is empty, singleton, or infinite.
  - **EUF is convex:** if  $F \Rightarrow x_1 = y_1 \vee \dots \vee x_n = y_n$ , then  $F \wedge x_1 \neq y_1 \wedge \dots \wedge x_n \neq y_n$  is unsatisfiable. The congruence closure algorithm shows this by demonstrating for some  $i$  that  $F \wedge x_i \neq y_i$  is unsatisfiable, i.e., that  $F \Rightarrow x_i = y_i$  is valid.
  - **LIA (linear integer arithmetic) is not convex:** take  $F : \Leftrightarrow 1 \leq x \wedge x \leq 2 \wedge y = 1 \wedge z = 2$ ; then  $F$  implies  $x = y \vee x = z$  but neither  $x = y$  nor  $x = z$ .

## Non-Convex Theories

How to combine with a non-convex theory  $T_i$ ?

- We may infer in  $T_i$  from formula  $F_i$  only a disjunction  $x_1 = y_1 \vee \dots \vee x_n = y_n$ .
  - But not any equality  $x_i = y_i$  of this disjunction.
- However, this disjunction can be made minimal (strongest).
  - Start with the disjunction of all possible variable equalities.
  - If it cannot be inferred, no smaller disjunction can be inferred either.
  - Otherwise, strip every  $x_i = y_i$  if this keeps the disjunction inferred.
- For each remaining  $x_i = y_i$ , recursively call `NELSONOPPEN( $F \wedge x_i = y_i$ )`.
  - Return “true” if any call returns “true” and “false”, otherwise.

Thus the Nelson-Oppen method is also applicable to non-convex theories (but with generally much greater complexity).