CSE4051: Program Verification

Combining Multiple Theories

2025 Fall

Woosuk Lee

Need for Combination

- In software verification, formulas like the following one arise:

$$a = b + 2 \wedge A = write(B, a+1, 4) \wedge (read(A, b+3) = 2 \vee f(a-1) \neq f(b+1))$$

- Here reasoning is needed over
  - The theory of linear arithmetic ($T_{\mathbb{Z}}$)
  - The theory of arrays (T_A)
  - The theory of equality with uninterpreted functions (T_E)
- Remember that we only consider quantifier-free conjunctions of literals.
- Given theory solver for the three individual theories, can we combine them to obtain one for ($T_{\mathbb{Z}} \cup T_A \cup T_E$)?

Nelson-Oppen Combination Method

- Under certain conditions, the Nelson-Oppen combination method gives a positive answer.

Motivating Example

- Consider the following conjunction of formulae

$$
\begin{aligned}
f(f(x)-f(y)) &= a \\
f(0) &= a+2 \\
x &= y
\end{aligned}
$$

- There are two theories involved: $T_{\mathbb{R}}$ and T_E
- **FIRST STEP**: purify each literal so that it belongs to a single theory

$$
f(f(x)-f(y)) = a \implies \begin{aligned} f(e_1) &= a \\ e_1 &= f(x)-f(y) \end{aligned} \implies \begin{aligned} f(e_1) &= a \\ e_1 &= e_2 - e_3 \\ e_2 &= f(x) \\ e_3 &= f(y) \end{aligned}
$$

Motivating Example

- **SECOND STEP**: check satisfiability and exchange entailed equalities

$$T_E \qquad\qquad\qquad T_{\mathbb{R}}$$

$$\begin{aligned} f(e_1) &= a & e_2 - e_3 &= e_1 \\ f(x) &= e_2 & e_4 &= 0 \\ f(y) &= e_3 & e_5 &= a+2 \\ f(e_4) &= e_5 \\ x &= y \end{aligned}$$

- The two solvers only share $e_1, e_2, e_3, e_4, e_5, a$.
- To merge the two models into a single one, the solvers have to agree on equalities between shared constants (interface equalities)
- This can be done by exchanging entailed interface equalities.

Motivating Example

- **SECOND STEP**: check satisfiability and exchange entailed equalities

T_E			$T_{\mathbb{R}}$		
$f(e_1)$	$=$	a	$e_2 - e_3$	$=$	e_1
$f(x)$	$=$	e_2	e_4	$=$	0
$f(y)$	$=$	e_3	e_5	$=$	$a+2$
$f(e_4)$	$=$	e_5	e_2	$=$	e_3
x	$=$	y			

- The two solvers only share $e_1, e_2, e_3, e_4, e_5, a$.
- T_E-Solver says SAT, and $T_{\mathbb{R}}$-Solver says SAT
- T_E-Solver says $e_2 = e_3$.

Motivating Example

- **SECOND STEP**: check satisfiability and exchange entailed equalities

T_E				$T_{\mathbb{R}}$		
$f(e_1)$	$=$	a		$e_2 - e_3$	$=$	e_1
$f(x)$	$=$	e_2		e_4	$=$	0
$f(y)$	$=$	e_3		e_5	$=$	$a + 2$
$f(e_4)$	$=$	e_5		e_2	$=$	e_3
x	$=$	y				
e_1	$=$	e_4				

- The two solvers only share $e_1, e_2, e_3, e_4, e_5, a$.
- T_E-Solver says SAT, and $T_{\mathbb{R}}$-Solver says SAT
- $T_{\mathbb{R}}$-Solver says $e_1 = e_4$.

Motivating Example

- **SECOND STEP**: check satisfiability and exchange entailed equalities

T_E			$T_{\mathbb{R}}$		
$f(e_1)$	$=$	a	$e_2 - e_3$	$=$	e_1
$f(x)$	$=$	e_2	e_4	$=$	0
$f(y)$	$=$	e_3	e_5	$=$	$a+2$
$f(e_4)$	$=$	e_5	e_2	$=$	e_3
x	$=$	y	a	$=$	e_5
e_1	$=$	e_4			

- The two solvers only share $e_1, e_2, e_3, e_4, e_5, a$.
- T_E-Solver says SAT, and $T_{\mathbb{R}}$-Solver says SAT
- T_E-Solver says $a = e_5$.

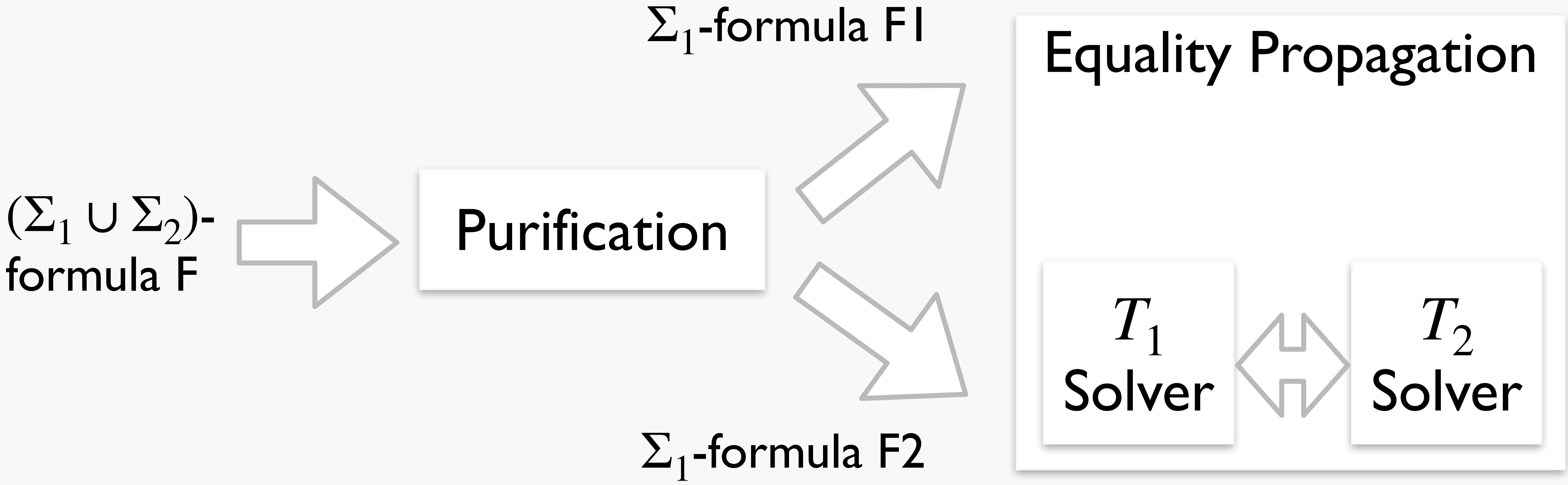
Motivating Example

- **SECOND STEP**: check satisfiability and exchange entailed equalities

T_E	$T_{\mathbb{R}}$
$f(e_1) = a$	$e_2 - e_3 = e_1$
$f(x) = e_2$	$e_4 = 0$
$f(y) = e_3$	$e_5 = a + 2$
$f(e_4) = e_5$	$e_2 = e_3$
$x = y$	$a = e_5$
$e_1 = e_4$	

- The two solvers only share $e_1, e_2, e_3, e_4, e_5, a$.
- T_E-Solver says SAT, and $T_{\mathbb{R}}$-Solver says **UNSAT.**
- Therefore, the formula is **UNSAT.**

Nelson-Oppen Algorithm



Purification

- Transforms a quantifier-free conjunctive formula F into two quantifier-free conjunctive formula, a Σ_1 formula F_1 and a Σ_2 formula F_2 such that $F = F_1 \wedge F_2$ and F_1 in T_1 and F_2 in T_2.
- Repeat until
  - If $s \in \Sigma_i$ (where $i = 1$ or 2) and $t \notin \Sigma_i$, and w is a fresh variable:

$$F[s = t] \quad \Longrightarrow \quad F[w = t] \ \wedge \ w = s$$

  - If function $f \in \Sigma_i$ (where $i = 1$ or 2) and $t \notin \Sigma_i$, and w is a fresh variable: (similarly for predicates)

$$F[f(t_1, \ \ldots, \ t, \ \ldots, \ t_n)] \quad \Longrightarrow \quad F[f(t_1, \ \ldots, \ w, \ \ldots, \ t_n)] \ \wedge \ w = t$$

Example I

- Consider $\Sigma_E \cup \Sigma_{\mathbb{Z}}$- formula

$$F : 1 \leq x \wedge x \leq 2 \wedge f(x) \neq f(1) \wedge f(x) \neq f(2)$$

- Since $f \in \Sigma_E$ and $1 \in \Sigma_{\mathbb{Z}}$, replace $f(1)$ by $f(w_1)$ and add $w_1 = 1$

$$1 \leq x \wedge x \leq 2 \wedge f(x) \neq f(w_1) \wedge f(x) \neq f(2) \wedge w_1 = 1$$

- Since $f \in \Sigma_E$ and $2 \in \Sigma_{\mathbb{Z}}$, similarly add $w_2 = 2$

$$1 \leq x \wedge x \leq 2 \wedge f(x) \neq f(w_1) \wedge f(x) \neq f(w_2) \wedge w_1 = 1 \wedge w_2 = 2$$

- Done. Construct $\Sigma_{\mathbb{Z}}$ formula

$$F_{\mathbb{Z}} : 1 \leq x \wedge x \leq 2 \wedge w_1 = 1 \wedge w_2 = 2$$

and Σ_E formula

$F_E : f(x) \neq f(w_1) \wedge f(x) \neq f(w_2)$. $F_{\mathbb{Z}}$ and F_E share variables x, w_1, and w_2.

Example 2

- Consider $\Sigma_E \cup \Sigma_{\mathbb{Z}}$- formula

$$F: \ f(x) = x + y \ \wedge \ x \leq y + z \ \wedge \ x + z \leq y \ \wedge \ y = 1 \ \wedge \ f(x) \neq f(2)$$

- Since $f \in \Sigma_E$ and $+ \in \Sigma_{\mathbb{Z}}$,

$$f(x) = x + y \implies w_1 = x + y \wedge w_1 = f(x)$$

- Since $f \in \Sigma_E$ and $2 \in \Sigma_{\mathbb{Z}}$,

$$f(x) \neq f(2) \implies f(x) \neq f(w_2) \wedge w_2 = 2$$

- Done. Construct $\Sigma_{\mathbb{Z}}$ formula

$$F_{\mathbb{Z}} : w_1 = x + y \wedge x \leq y + z \wedge x + z \leq y \wedge y = 1 \wedge w_2 = 2$$

and Σ_E formula

$F_E : w_1 = f(x) \wedge f(x) \neq f(w_2)$. $F_{\mathbb{Z}}$ and F_E share variables x and w_2.

Nelson-Oppen Algorithm

function Nelson-Oppen(F) {

1: Purify F into $F_1 \wedge F_2$

2: r_1 := Run T_1 solver on F_1

3: r_2 := Run T_2 solver on F_2

4: **if** r_1 = UNSAT or r_2 = UNSAT **then return** UNSAT

5: **if** there exists shared variables x, y such that

6: $F_i \Rightarrow x = y$ but F_j does not for $i, j \in \{1,2\}$

then

7: $F_j := F_j \wedge x = y$

8: Goto line 2

9: return SAT

Example

- Consider $\Sigma_E \cup \Sigma_{\mathbb{Q}}$-formula

  $$F : \ f(f(x) - f(y)) \neq f(z) \ \wedge \ x \leq y \ \wedge \ y + z \leq x \ \wedge \ 0 \leq z$$

- **Phase I Purification :** Purify F into

  $$F_{\mathsf{E}} : \ f(w) \neq f(z) \ \wedge \ u = f(x) \ \wedge \ v = f(y)$$

  and

  $$F_{\mathbb{Q}} : \ x \leq y \ \wedge \ y + z \leq x \ \wedge \ 0 \leq z \ \wedge \ w = u - v$$

  with shared variables x, y, z, u, v, w.

Example

- **Phase 2 Equality Propagation**

T_E

$$f(w) \neq f(z)$$
$$u = f(x)$$
$$v = f(y)$$
$$x = y$$

$T_{\mathbb{Q}}$

$$x \leq y$$
$$y + z \leq x$$
$$0 \leq z$$
$$w = u - v$$

- Shared variables: x, y, z, u, v, w.
- T_E-Solver says SAT, and $T_{\mathbb{Q}}$-Solver says SAT
- $T_{\mathbb{Q}}$-Solver says $x = y$

Example

- **Phase 2 Equality Propagation**

T_E	$T_{\mathbb{Q}}$
	$x \leq y$
$f(w) \neq f(z)$	$y + z \leq x$
$u = f(x)$	$0 \leq z$
$v = f(y)$	$w = u - v$
$x = y$	$u = v$

- Shared variables: x, y, z, u, v, w.
- T_E-Solver says SAT, and $T_{\mathbb{Q}}$-Solver says SAT
- T_E-Solver says $u = v$

Example

- **Phase 2 Equality Propagation**

T_E

$$f(w) \neq f(z)$$
$$u = f(x)$$
$$v = f(y)$$
$$x = y$$
$$z = w$$

$T_{\mathbb{Q}}$

$$x \leq y$$
$$y + z \leq x$$
$$0 \leq z$$
$$w = u - v$$
$$u = v$$

- Shared variables: x, y, z, u, v, w.
- T_E-Solver says SAT, and $T_{\mathbb{Q}}$-Solver says SAT
- $T_{\mathbb{Q}}$-Solver says $z = w$

Example

- **Phase 2 Equality Propagation**

T_E

$$f(w) \neq f(z)$$
$$u = f(x)$$
$$v = f(y)$$
$$x = y$$
$$z = w$$

$T_{\mathbb{Q}}$

$$x \leq y$$
$$y + z \leq x$$
$$0 \leq z$$
$$w = u - v$$
$$u = v$$

- Shared variables: x, y, z, u, v, w.
- T_E-Solver says SAT, and $T_{\mathbb{Q}}$-Solver says SAT
- $T_{\mathbb{Q}}$-Solver says $z = w$

Example

- **Phase 2 Equality Propagation**

T_E

$$f(w) \neq f(z)$$
$$u = f(x)$$
$$v = f(y)$$
$$x = y$$
$$z = w$$

$T_{\mathbb{Q}}$

$$x \leq y$$
$$y + z \leq x$$
$$0 \leq z$$
$$w = u - v$$
$$u = v$$

- Shared variables: x, y, z, u, v, w.
- T_E-Solver says **UNSAT**.
- Therefore, the formula is **UNSAT**.

Nelson-Oppen Restrictions

- Two theories can be combined when
  - Both are decidable, quantifier-free conjunctive fragments
  - Equality (=) is the only symbol in the intersection of their signatures.
  - Both are *stably infinite*

    (not covered in this lecture; see textbook if you feel interested)

- The algorithm shown in this lecture is the *deterministic* version of Nelson-Oppen method. Only applicable when theories are *convex*
  - The *nondeterministic* version is applicable when theories are not convex

    (not covered in this lecture; see textbook if you feel interested)

Summary

- Nelson-Oppen Method
- Purification
- Equality propagation
- Some parts borrowed from

  Albert Oliveras, SMT Theory and DPLL(T), 1st SAT/SMT solver summer school 2011