Automated Reasoning and Program Verification

Laura Kovács
TU Vienna

Problems – Where are we now?

- ✓ Deciding theory: Check satisfiability of a set of *literals* in $\mathcal{T}_E$, $\mathcal{T}_A$, and $\mathcal{T}_Q$.
- ✓ Put together theory reasoning and SAT solving

Problems – Where are we now?

- ✓ Deciding theory: Check satisfiability of a set of *literals* in $\mathcal{T}_E$, $\mathcal{T}_A$, and $\mathcal{T}_Q$.
- ✓ Put together theory reasoning and SAT solving
- ? What about combination of theories: Given decision procedures for theories, how can we build a decision procedure for formulas using several theories?

Problems – Where are we now?

- ✓ Deciding theory: Check satisfiability of a set of *literals* in $\mathcal{T}_E$, $\mathcal{T}_A$, and $\mathcal{T}_Q$.
- ✓ Put together theory reasoning and SAT solving
- ? What about combination of theories: Given decision procedures for theories, how can we build a decision procedure for formulas using several theories?

We next study satisfiability of formulas in combination of theories!

Outline

Reasoning with a union of theories

Given:

- Theories $T_1, \ldots, T_n$ in disjoint signatures $\Sigma_1, \ldots, \Sigma_n$;
- Quantifier-free formula F in $\Sigma_1 \cup \ldots \cup \Sigma_n$.

Reasoning with a union of theories

Given:

- Theories $T_1, \ldots, T_n$ in disjoint signatures $\Sigma_1, \ldots, \Sigma_n$;
- Quantifier-free formula F in $\Sigma_1 \cup \ldots \cup \Sigma_n$.

Question:

- Is F satisfiable in $T_1 \cup \ldots \cup T_n$?

Reasoning with a combination of theories

One can also formulate a similar problem when theories are given as classes of interpretations.

Reasoning with a combination of theories

One can also formulate a similar problem when theories are given as classes of interpretations.

In this case satisfiability means finding an interpretation of $\Sigma_1 \cup \ldots \cup \Sigma_n$ such that its restriction to each Σ_i belongs to the class of interpretations of T_i.

Reasoning with a combination of theories

One can also formulate a similar problem when theories are given as classes of interpretations.

In this case satisfiability means finding an interpretation of $\Sigma_1 \cup \ldots \cup \Sigma_n$ such that its restriction to each Σ_i belongs to the class of interpretations of T_i.

We will use the term combination,
and mean reasoning with a union/combination of theories.

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(\mathit{read}(\mathit{write}(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays
- uninterpreted functions

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays
- uninterpreted functions

Sorts/types?

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays
- uninterpreted functions

Sorts/types?

- $x, y : int$

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays
- uninterpreted functions

Sorts/types?

- $x, y : int$
- a : array[int,int]

Example

Is this set of formulas satisfiable?

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

- linear arithmetic
- arrays
- uninterpreted functions

Sorts/types?

- $x, y : int$
- $a : \text{array[int,int]}$
- $f : int \rightarrow s$

How to solve?

- Use DPLL(T). This allows one to reduce the satisfiability problem for quantifier-free formulas to the satisfiability problem for conjunctions of literals.

How to solve?

- Use DPLL(T). This allows one to reduce the satisfiability problem for quantifier-free formulas to the satisfiability problem for conjunctions of literals.

Checking satisfiability of a conjunction of literals in a combination of theories?

How to solve?

- Use DPLL(T). This allows one to reduce the satisfiability problem for quantifier-free formulas to the satisfiability problem for conjunctions of literals.

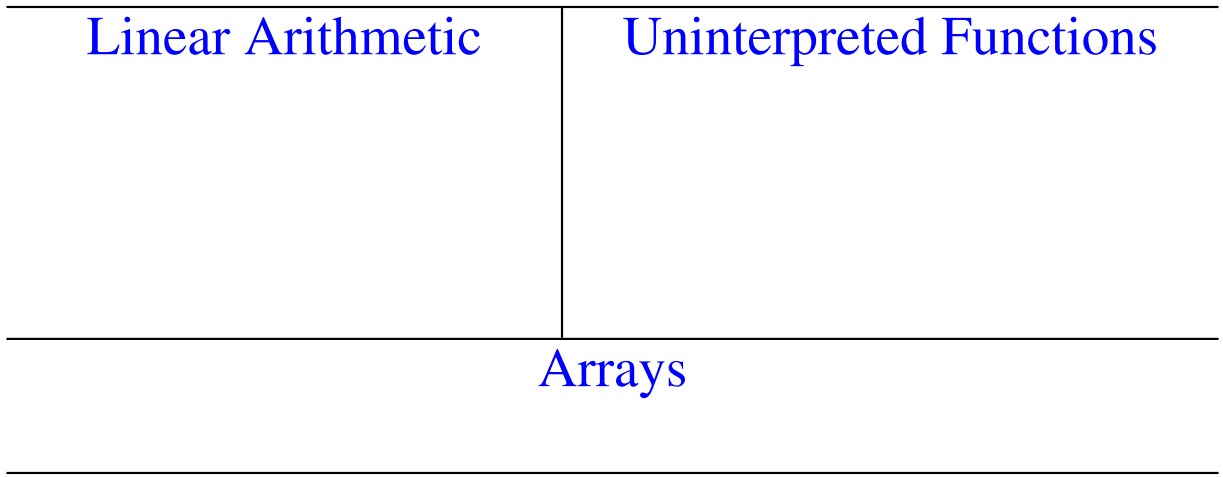
Checking satisfiability of a conjunction of literals in a combination of theories?

Idea:

- Separate reasoning in various theories
- Make reasoners exchange equalities

Separating Reasoning

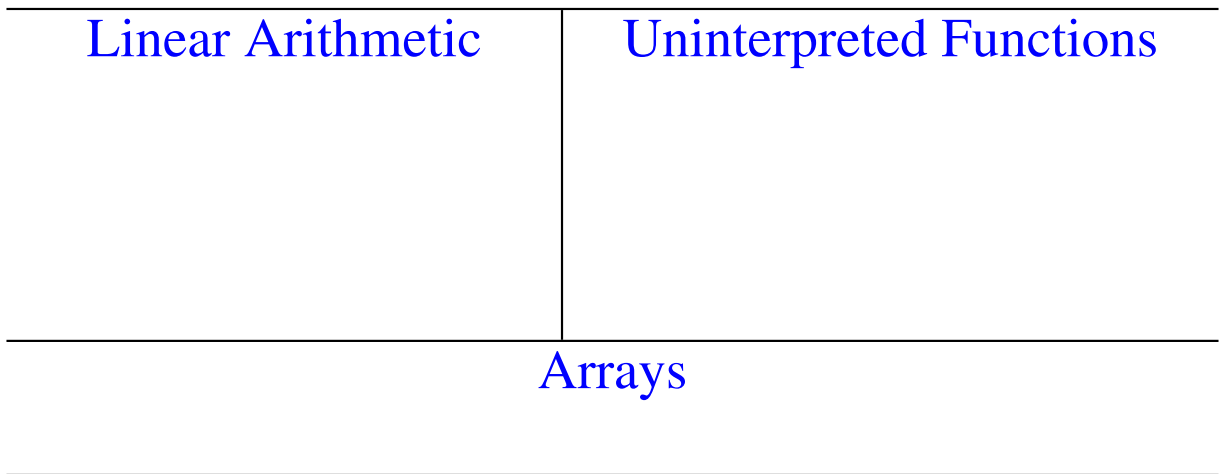
$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$



Separating Reasoning

$x + 2 = y$

$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$



Separating Reasoning

$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, 3), y - 2)) \neq f(c_1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, 3), y - 2)) \neq f(c_1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, 3), c_2)) \neq f(c_1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	
$c_1 = y - x + 1$	
$c_2 = y - 2$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, 3), c_2)) \neq f(c_1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	
$c_1 = y - x + 1$	
$c_2 = y - 2$	
$c_3 = 3$	
Arrays	

Separating Reasoning

$$f(read(write(a, x, c_3), c_2)) \neq f(c_1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	
$c_1 = y - x + 1$	
$c_2 = y - 2$	
$c_3 = 3$	

Arrays

$$c_4 = read(write(a, x, c_3), c_2)$$

Separating Reasoning

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

Separating Reasoning

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	$f(c_4) \neq f(c_1)$
$c_1 = y - x + 1$	
$c_2 = y - 2$	
$c_3 = 3$	
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

- Each step obviously preserves satisfiability; moreover every model of the modified set of literals is also a model of the original set.

Separating Reasoning

procedure *SeparatingReasoning*(F)
input: formula $F \in \mathcal{T}_1 \cup \cdots \cup \mathcal{T}_n$
output: formulas $F_1 \in \mathcal{T}_1, \ldots, F_n \in \mathcal{T}_n$ s.t. $F \equiv F_1 \wedge \cdots \wedge F_n$
assumptions: theory signatures $\Sigma_1, \ldots, \Sigma_n$ are disjoint
parameters: function $head(t)$ returning the root symbol of a term t
begin
  repeat as long as possible
    if $f \in \Sigma_i$ and $head(t) \in \Sigma_j$ with $i \neq j$:
      rewrite $F[f(t_1, \ldots, t, \ldots, t_m)]$ into $F[f(t_1, \ldots, c, \ldots, t_m)] \wedge c = t$,
      where c is a fresh new variable
    if $p \in \Sigma_i$ and $head(t) \in \Sigma_j$ with $i \neq j$:
      rewrite $F[p(t_1, \ldots, t, \ldots, t_m)]$ into $F[p(t_1, \ldots, c, \ldots, t_m)] \wedge c = t$,
      where c is a fresh new variable
    if $head(s) \in \Sigma_i$ and $head(t) \in \Sigma_j$ with $i \neq j$:
      rewrite $F[s = t]$ into $F[s = c] \wedge c = t$,
      where c is a fresh new variable
  end repeat
return modified F as $F_1 \wedge \cdots \wedge F_n$, with each $F_i \in \mathcal{T}_i$

Separating Reasoning

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	$f(c_4) \neq f(c_1)$
$c_1 = y - x + 1$	
$c_2 = y - 2$	
$c_3 = 3$	
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

- Each step obviously preserves satisfiability; moreover every model of the modified set of literals is also a model of the original set.

Separating Reasoning

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

- Each step obviously preserves satisfiability; moreover every model of the modified set of literals is also a model of the original set.
- In this example every "separated" subset of literals is satisfiable.

Separating Reasoning

$$x + 2 = y$$
$$f(read(write(a, x, 3), y - 2)) \neq f(y - x + 1)$$

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

- Each step obviously preserves satisfiability; moreover every model of the modified set of literals is also a model of the original set.
- In this example every "separated" subset of literals is satisfiable.
- But this does not mean that the set of all literals is satisfiable.

System to Solve

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$	$f(c_4) \neq f(c_1)$
$c_1 = y - x + 1$	
$c_2 = y - 2$	
$c_3 = 3$	
Arrays	
$c_4 = read(write(a, x, c_3), c_2)$	

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	**Equalities**
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

Exchanging Equalities

<table>
<tr>
<td>Linear Arithmetic
$y = x + 2$
$c_1 = y - x + 1$
$c_2 = y - 2$
$c_3 = 3$</td>
<td>Uninterpreted Functions
$f(c_4) \neq f(c_1)$</td>
</tr>
<tr>
<td>Arrays
$c_4 = read(write(a, x, c_3), c_2)$</td>
<td>Equalities
$c_2 = x$
$c_4 = c_3$
$c_1 = c_3$</td>
</tr>
</table>

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	**Equalities**
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

- Congruence closure returns “unsatisfiable”.

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	**Equalities**
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

- Congruence closure returns “unsatisfiable”.
- All derived equalities are implied by the initial set of literals. Therefore, the initial set is unsatisfiable.

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	**Equalities**
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

- Congruence closure returns “unsatisfiable”.
- All derived equalities are implied by the initial set of literals. Therefore, the initial set is unsatisfiable.
- Therefore the original set (without extra variables) is unsatisfiable, too.

Exchanging Equalities

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 1$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	**Equalities**
$c_4 = read(write(a, x, c_3), c_2)$	$c_2 = x$ $c_4 = c_3$ $c_1 = c_3$

- Congruence closure returns "unsatisfiable".
- All derived equalities are implied by the initial set of literals. Therefore, the initial set is unsatisfiable.
- Therefore the original set (without extra variables) is unsatisfiable, too.

Not all theories can be combined (easily) in this way!

Not all theories can be combined in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;

Not all theories can be combined in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
   - A theory $\mathcal{T}_i$ with signature Σ_i is stably infinite if for every satisfiable formula $F \in \mathcal{T}_i$ there exists some $\mathcal{T}$-interpretation such that:
     - $I \models F$, and
     - I has a domain of infinite cardinality.

Not all theories can be combined in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
   - A theory $\mathcal{T}_i$ with signature Σ_i is stably infinite if for every satisfiable formula $F \in \mathcal{T}_i$ there exists some $\mathcal{T}$-interpretation such that:
     - $I \models F$, and
     - I has a domain of infinite cardinality.

$\mathcal{T}_E, \mathcal{T}_A, \mathcal{T}_Q$ are stably infinite.

Not all theories can be combined in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
   - A theory $\mathcal{T}_i$ with signature Σ_i is stably infinite if for every satisfiable formula $F \in \mathcal{T}_i$ there exists some $\mathcal{T}$-interpretation such that:
     - $I \models F$, and
     - I has a domain of infinite cardinality.

$\mathcal{T}_E, \mathcal{T}_A, \mathcal{T}_Q$ are stably infinite.

Problem with combining theories, as described, that are not stably infinite. Why?

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.
   - A theory $\mathcal{T}_i$ is convex if for every formula $F \in \mathcal{T}_i$ such that F is a conjunction of $\mathcal{T}_i$-literals:

$$\text{if } F \to \bigvee_{j=1}^{k} (u_j = v_j) \quad \text{then} \quad F \to u_j = v_j \text{ for some } j \in \{1, \ldots, k\}.$$

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.
   - A theory $\mathcal{T}_i$ is convex if for every formula $F \in \mathcal{T}_i$ such that F is a conjunction of $\mathcal{T}_i$-literals:

$$\text{if } F \to \bigvee_{j=1}^{k} (u_j = v_j) \quad \text{then} \quad F \to u_j = v_j \text{ for some } j \in \{1, \ldots, k\}.$$

Is $\mathcal{T}_{\mathbb{Z}}$ convex?

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.
   - A theory $\mathcal{T}_i$ is convex if for every formula $F \in \mathcal{T}_i$ such that F is a conjunction of $\mathcal{T}_i$-literals:

$$\text{if } F \to \bigvee_{j=1}^{k} (u_j = v_j) \quad \text{then} \quad F \to u_j = v_j \text{ for some } j \in \{1, \ldots, k\}.$$

Is $\mathcal{T}_{\mathbb{Z}}$ convex? No.

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.
   - A theory $\mathcal{T}_i$ is convex if for every formula $F \in \mathcal{T}_i$ such that F is a conjunction of $\mathcal{T}_i$-literals:

$$\text{if } F \to \bigvee_{j=1}^{k} (u_j = v_j) \quad \text{then} \quad F \to u_j = v_j \text{ for some } j \in \{1, \ldots, k\}.$$

Is $\mathcal{T}_{\mathbb{Z}}$ convex? No.

Is $\mathcal{T}_A$ convex?

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.
   - A theory $\mathcal{T}_i$ is convex if for every formula $F \in \mathcal{T}_i$ such that F is a conjunction of $\mathcal{T}_i$-literals:

$$\text{if } F \to \bigvee_{j=1}^{k} (u_j = v_j) \quad \text{then} \quad F \to u_j = v_j \text{ for some } j \in \{1, \ldots, k\}.$$

Is $\mathcal{T}_{\mathbb{Z}}$ convex? No.

Is $\mathcal{T}_A$ convex? No.

$\mathcal{T}_E, \mathcal{T}_Q$ are convex.

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.

Exchange of equalities between $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

- for a convex theory $\mathcal{T}_i$, its decision procedure discovers a new equality $u = v$, for shared u, v;

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.

Exchange of equalities between $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

- for a convex theory $\mathcal{T}_i$, its decision procedure discovers a new equality $u = v$, for shared u, v; Pass this new equality to the decision procedures of the other theories $\mathcal{T}_j$;

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.

Exchange of equalities between $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

- for a convex theory $\mathcal{T}_i$, its decision procedure discovers a new equality $u = v$, for shared u, v; Pass this new equality to the decision procedures of the other theories $\mathcal{T}_j$;
- for a non-convex theory $\mathcal{T}_i$, its decision procedure discovers a disjunction of new equalities $\bigvee_k u_k = v_k$, for shared u_k, v_k;
  - Split the disjunction and exchange equalities along multiple branches: for each $u_k = v_k$, pass the equality to the decision procedures of the other theories $\mathcal{T}_j$.

Not all theories can be combined EASILY in this way!

Restrictions on theories $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

1. Theories have disjoint signatures;
2. Each theory $\mathcal{T}_i$ is stably infinite;
3. Each theory $\mathcal{T}_i$ is convex.

Exchange of equalities between $\mathcal{T}_1, \ldots, \mathcal{T}_n$:

- for a convex theory $\mathcal{T}_i$, its decision procedure discovers a new equality $u = v$, for shared u, v; Pass this new equality to the decision procedures of the other theories $\mathcal{T}_j$;
- for a non-convex theory $\mathcal{T}_i$, its decision procedure discovers a disjunction of new equalities $\bigvee_k u_k = v_k$, for shared u_k, v_k;
  - Split the disjunction and exchange equalities along multiple branches: for each $u_k = v_k$, pass the equality to the decision procedures of the other theories $\mathcal{T}_j$.

    For efficiency, work with minimal disjunctions.

Combination of theories: The Nelson-Oppen method

The Nelson-Oppen method (1979):

- for reasoning in combination of stably infinite theories with disjoint signatures;
- by separating reasoning and exchange of equalities;

Combination of theories: The Nelson-Oppen method

The Nelson-Oppen method (1979):

- for reasoning in combination of stably infinite theories with disjoint signatures;
- by separating reasoning and exchange of equalities;
- for convex theories $\mathcal{T}_i$ with decision procedures in P, the Nelson-Oppen method is in P;
- for theories $\mathcal{T}_i$ with decision procedures in NP, the Nelson-Oppen method is in NP.

Outline

Models of Satisfiable Sets

<table>
<tr>
<td>Linear Arithmetic
$\begin{aligned} y &= x + 2 \\ c_1 &= y - x + 2 \\ c_2 &= y - 2 \\ c_3 &= 3 \end{aligned}$</td>
<td>Uninterpreted Functions
$f(c_4) \neq f(c_1)$</td>
</tr>
<tr>
<td>Arrays
$c_4 = read(write(a, x, c_3), c_2)$</td>
<td>Equalities</td>
</tr>
</table>

Models of Satisfiable Sets

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays	Equalities
$c_4 = read(write(a, x, c_3), c_2)$	

Models of Satisfiable Sets

<table>
<tr>
<td>Linear Arithmetic
$\begin{aligned} y &= x + 2 \\ c_1 &= y - x + 2 \\ c_2 &= y - 2 \\ c_3 &= 3 \end{aligned}$</td>
<td>Uninterpreted Functions
$f(c_4) \neq f(c_1)$</td>
</tr>
<tr>
<td>Arrays
$c_4 = read(write(a, x, c_3), c_2)$</td>
<td>Equalities
$c_2 = x$</td>
</tr>
</table>

Models of Satisfiable Sets

Linear Arithmetic $y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	Uninterpreted Functions $f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$

Models of Satisfiable Sets

Linear Arithmetic $y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	Uninterpreted Functions $f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$

Models of Satisfiable Sets

<table>
<tr>
<td>Linear Arithmetic
$y = x + 2$
$c_1 = y - x + 2$
$c_2 = y - 2$
$c_3 = 3$</td>
<td>Uninterpreted Functions
$f(c_4) \neq f(c_1)$</td>
</tr>
<tr>
<td>Arrays
$c_4 = read(write(a, x, c_3), c_2)$</td>
<td>Equalities
$c_2 = x$
$c_4 = c_3$</td>
</tr>
</table>

Models of Satisfiable Sets

Linear Arithmetic $y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	Uninterpreted Functions $f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$ $c_4 = c_3$

Models of Satisfiable Sets

Linear Arithmetic $y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	Uninterpreted Functions $f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$ $c_4 = c_3$

Models of Satisfiable Sets

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$ $c_4 = c_3$ $c_4 = 3$

Models of Satisfiable Sets

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$ $c_4 = c_3$ $c_4 = 3$ $c_1 = 4$

Models of Satisfiable Sets

Linear Arithmetic	Uninterpreted Functions
$y = x + 2$ $c_1 = y - x + 2$ $c_2 = y - 2$ $c_3 = 3$	$f(c_4) \neq f(c_1)$
Arrays $c_4 = read(write(a, x, c_3), c_2)$	Equalities $c_2 = x$ $c_4 = c_3$ $c_4 = 3$ $c_1 = 4$

Take models of:

- the literals in every theory plus the set of derived equalities.

Running an SMT solver

The most efficient SMT solver: Z3, http://rise4fun.com/Z3.

Running an SMT solver

The most efficient SMT solver: Z3, `http://rise4fun.com/Z3`.

SMTLib2 input format.

Running an SMT solver

The most efficient SMT solver: Z3, `http://rise4fun.com/Z3`.

SMTLib2 input format.

Easy to use.

Problems – Where are we now?

- ✓ Deciding theory: Check satisfiability of a set of *literals* in $\mathcal{T}_E$, $\mathcal{T}_A$, and $\mathcal{T}_Q$.
- ✓ SMT and non-unit formulas: Put together theory reasoning and SAT solving
- ✓ Reasoning in combination of theories: Given decision procedures for theories, we can build a decision procedure for the combination of these theories.