On Shostak's Combination of Decision Procedures

H. Rueß, N. Shankar, A. Tiwari

{ruess,shankar,tiwari}@csl.sri.com

http://www.csl.sri.com/.

Computer Science Laboratory

SRI International

333 Ravenswood

Menlo Park, CA 94025

The Combination Problem

Verification conditions typically are in combination of many theories.

- Theory of equality
- Arithmetic constraints
- Lists, Arrays, Bitvectors, …

Examples.

- $x + 2 = y \vdash f(a[x := 3][y - 2]) = f(y - x + 1)$
- $f(y-1)-1 = y+1,\ f(x)+1 = x-1,\ x+1 = y \vdash \mathit{false}$
- $f(f(x) - f(y)) \neq f(z),\ y \leq x, y \geq x + z, z > 0 \vdash \mathit{false}$

West-Coast Theorem Proving

Theorem provers which rely heavily on decision procedures for automating proofs.

Historical.

- Stanford Pascal Verifier
- Boyer-Moore Theorem Prover
- Shostak's Theorem Prover (STP)

Current. (Outline)

- Simplify, Java/ESC
- Stanford Temporal Prover (STeP)
- Stanford Validity Checker (SVC)
- PVS

Backend Decision Procedures

- Decision procedures used as prover backend
- Logical context stored in a database
- Communication via *ask/tell* interface
- Decision procedure for equality in a combination of theories based on
  - Nelson and Oppen's [1979]
  - Shostak's [1984]
- These combination algorithms use variants of congruence closure algorithms

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Language: Signatures

A *signature*, Σ, is a finite set of

Function Symbols	:	$\Sigma_F = \{f, g, \ldots\}$
Predicate Symbols	:	$\Sigma_P = \{P, Q, \ldots\}$

along with an arity function $arity : \Sigma \mapsto \mathbb{N}$.

Function symbols with arity 0 are called *constants* and denoted by $a, b, \ldots$, with possible subscripts.

A countable set $\mathcal{V}$ of *variables* is assumed disjoint of Σ.

Language: Terms

The set $\mathcal{T}(\Sigma, \mathcal{V})$ of *terms* is the smallest set s.t.

- $\mathcal{V} \subset \mathcal{T}(\Sigma, \mathcal{V})$, and
- $f(t_1, \ldots, t_n) \in \mathcal{T}(\Sigma, \mathcal{V})$ whenever $t_1, \ldots, t_n \in \mathcal{T}(\Sigma, \mathcal{V})$ and $arity(f) = n$.

The set of *ground* terms is defined as $\mathcal{T}(\Sigma, \emptyset)$.

Language: Atomic Formulas

An *atomic formula* is an expression of the form

$$P(t_1, \ldots, t_n)$$

where P is a predicate in Σ s.t. $arity(P) = n$ and $t_1, \ldots, t_n \ \in \ \mathcal{T}(\Sigma, \mathcal{V})$.

If $t_1, \ldots, t_n$ are ground terms, then $P(t_1, \ldots, t_n)$ is called a ground (atomic) formula.

Mostly, we assume a special binary predicate $=$ to be present in Σ.

Language: Logical Symbols

The set of *quantifier-free formula* (over Σ), $QFF(\Sigma, \mathcal{V})$, is the smallest set s.t.

- Every atomic formula is in $QFF(\Sigma, \mathcal{V})$,
- If $\phi \in QFF(\Sigma, \mathcal{V})$, then $\neg\phi \in QFF(\Sigma, \mathcal{V})$,
- If $\phi_1, \phi_2 \in QFF(\Sigma, \mathcal{V})$, then

$$\begin{array}{rcccl} \phi_1 & \wedge & \phi_2 & \in & QFF(\Sigma, \mathcal{V}) \\ \phi_1 & \vee & \phi_2 & \in & QFF(\Sigma, \mathcal{V}) \\ \phi_1 & \Rightarrow & \phi_2 & \in & QFF(\Sigma, \mathcal{V}) \\ \phi_1 & \Leftrightarrow & \phi_2 & \in & QFF(\Sigma, \mathcal{V}). \end{array}$$

An atomic formula or its negation is a *literal*.

Language: Sentence, Theory

The closure of $QFF(\Sigma, \mathcal{V})$ under existential ($\exists$) and universal ($\forall$) quantification defines the set of *(first-order) formulas*.

A *sentence* is a FO formula with no free variables.

A *(first-order) theory* $\mathcal{T}$ (over a signature Σ) is a set of (deductively closed) set of sentences (over Σ and $\mathcal{V}$).

A theory $\mathcal{T}$ is consistent if *false* $\notin \mathcal{T}$.

Due to completeness of first-order logic, we can identify a a FO theory $\mathcal{T}$ with the class of all models of $\mathcal{T}$.

Semantic Characterization

A model $\mathbb{A}$ is defined by a

- Domain A: set of elements
- Interpretation $f^{\mathbb{A}} : A^n \mapsto A$ for each $f \in \Sigma_F$ with $arity(f) = n$
- Interpretation $P^{\mathbb{A}} \subseteq A^n$ for each $P \in \Sigma_P$ with $arity(P) = n$
- Assignment $x^{\mathbb{A}} \in A$ for each variable $x \in \mathcal{V}$

A formula ϕ is true in a model $\mathbb{A}$ if it evaluates to true under the given interpretations over the domain A.

If all sentences in a $\mathcal{T}$ are true in a model $\mathbb{A}$, then $\mathbb{A}$ is a model for the theory $\mathcal{T}$.

Satisfiability and Validity

A formula $\phi(\vec{x})$ is *satisfiable* in a theory $\mathcal{T}$ if there is a model of $\mathcal{T} \cup \{\exists\vec{x}.\phi(\vec{x})\}$, i.e., there exists a model $\mathbb{M}$ for $\mathcal{T}$ in which ϕ evaluates to true, denoted by,

$$\mathbb{M} \models_{\mathcal{T}} \phi$$

This is also called $\mathcal{T}$-satisfiability.

A formula $\phi(\vec{x})$ is *valid* in a theory $\mathcal{T}$ if $\forall\vec{x}.\phi(\vec{x}) \in \mathcal{T}$, i.e., ϕ evaluates to true in every model $\mathbb{M}$ of $\mathcal{T}$. $\mathcal{T}$-validity is denoted by $\models_{\mathcal{T}} \phi$.

ϕ is $\mathcal{T}$-unsatisfiable if it is not the case that $\models_{\mathcal{T}} \phi$.

Getting Started

Checking validity of ϕ_0 in a theory $\mathcal{T}_0$:

$$
\begin{aligned}
&\equiv \quad \mathcal{T}_0\text{-satisfiability of } \neg\phi_0 \\
&\equiv \quad \mathcal{T}_0\text{-satisfiability of } \vec{Q}\vec{x}.\phi_1 \text{ (PNF)} \\
&\equiv \quad \mathcal{T}_1\text{-satisfiability of } \forall\vec{x}.\phi_1 \text{ (Skolemize)} \\
&\equiv \quad \mathcal{T}_1\text{-satisfiability of } \phi_3 \text{ (Instantiate)} \\
&\equiv \quad \mathcal{T}_1\text{-satisfiability of } \bigvee_i \psi_i \text{ (DNF)} \\
&\equiv \quad \mathcal{T}_1\text{-satisfiability of } \psi_i
\end{aligned}
$$

ψ_i: conjunction of literals

$\mathcal{T}_1$: $\mathcal{T}_0 \cup$ Theory of equality over UIFs

Pure Theory of Equality

$$\begin{array}{rcl} \Sigma & = & \Sigma_F \text{ (uninterpreted)} \\ \mathcal{T} & = & \text{Deductive closure of axioms of equality} \end{array}$$

Theorem 1 *Satisfiability of (quantifier-free) conjunction of literals is decidable in* $O(n\log(n))$*-time.*

Pure Theory of Equality

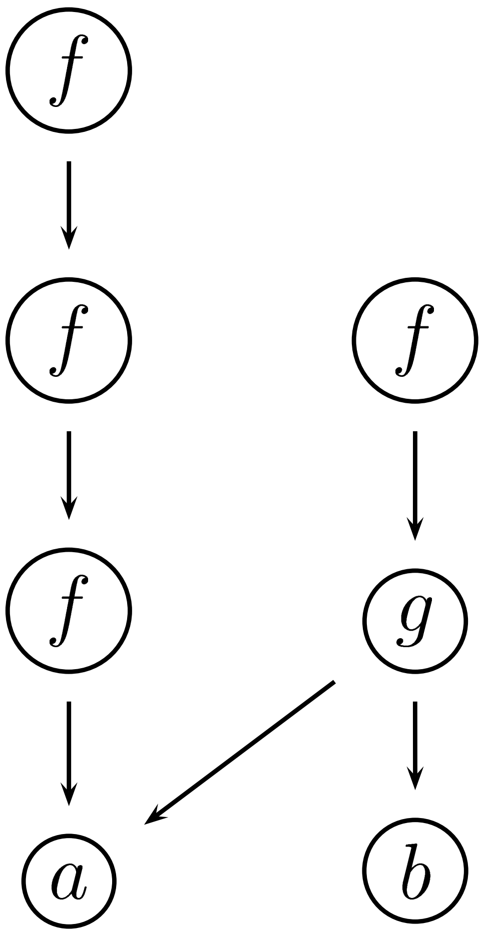
$$\begin{array}{rcl} \Sigma & = & \Sigma_F \text{ (uninterpreted)} \\ \mathcal{T} & = & \text{Deductive closure of axioms of equality} \end{array}$$

Theorem 1 *Satisfiability of (quantifier-free) conjunction of literals is decidable in* $O(n \log(n))$*-time.*

Example. Let $\Sigma = \{f^{(1)}, g^{(2)}, a^{(0)}, b^{(0)}\}$. Consider

$$a = gab \ \wedge \ fffa = fgab \ \wedge \ fffa \neq fa.$$

Illustration: D-rules and C-rules



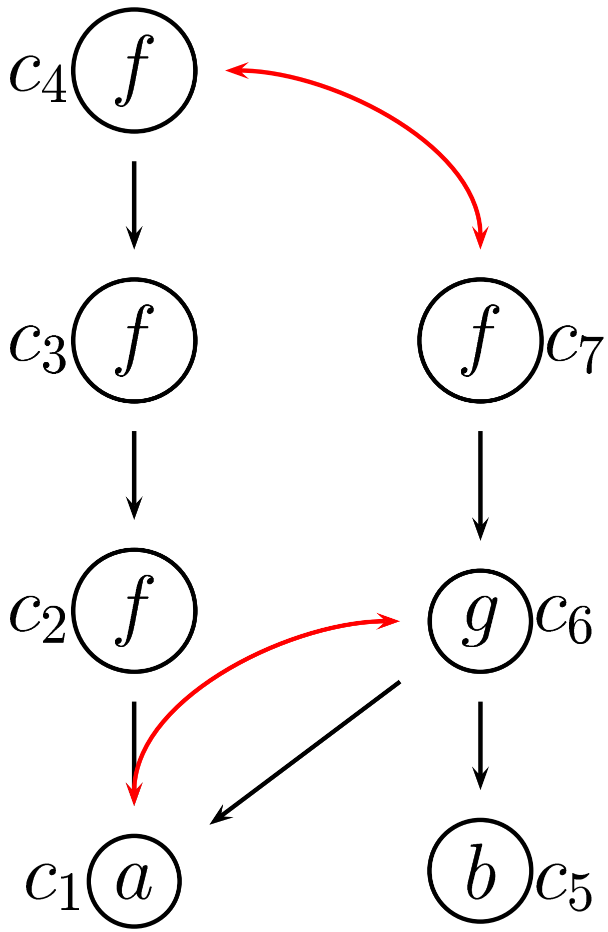
D-rules represent the term DAG:

$$\begin{array}{rclrcl} a & \to & c_1 & fc_1 & \to & c_2 \\ fc_2 & \to & c_3 & fc_3 & \to & c_4 \\ b & \to & c_5 & gc_1c_5 & \to & c_6 \\ fc_6 & \to & c_7 & & & \end{array}$$

Equations are represented as:

$$\begin{array}{ccccccc} a & = & gab & \wedge & fffa & = & fgab \\ \downarrow & & \downarrow & & \downarrow & & \downarrow \\ c_1 & = & c_6 & \wedge & c_4 & = & c_7 \end{array}$$

Illustration: D-rules and C-rules



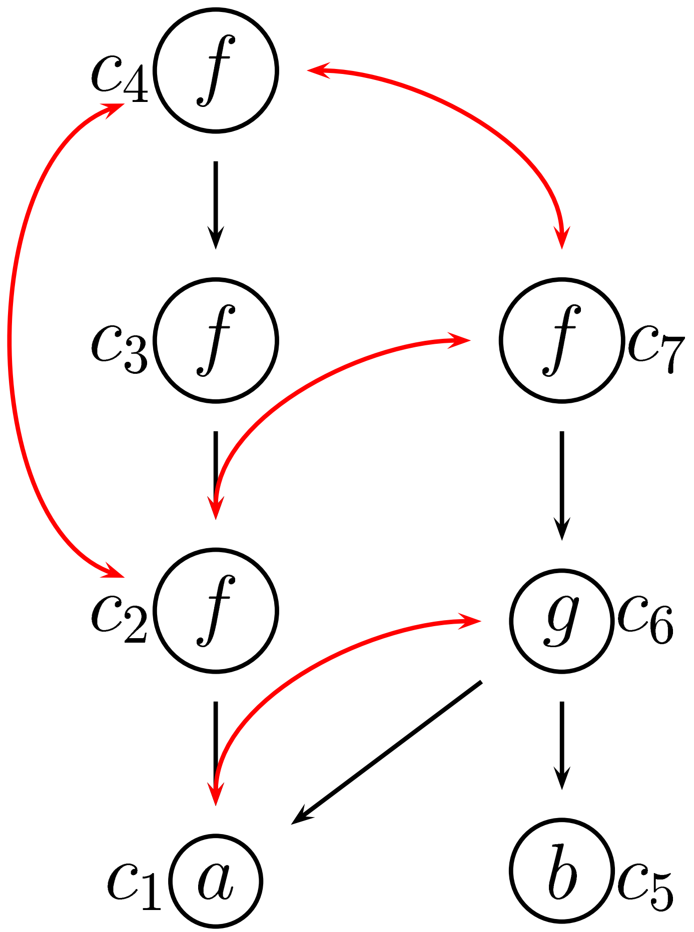
D-rules represent the term DAG:

$$
\begin{array}{rcl@{\qquad}rcl}
a & \to & c_1 & fc_1 & \to & c_2 \\
fc_2 & \to & c_3 & fc_3 & \to & c_4 \\
b & \to & c_5 & gc_1c_5 & \to & c_6 \\
fc_6 & \to & c_7 & & &
\end{array}
$$

C-rules represent an equivalence relation on vertices:

$$c_1 = c_6 \qquad c_4 = c_7$$

Illustration: D-rules and C-rules



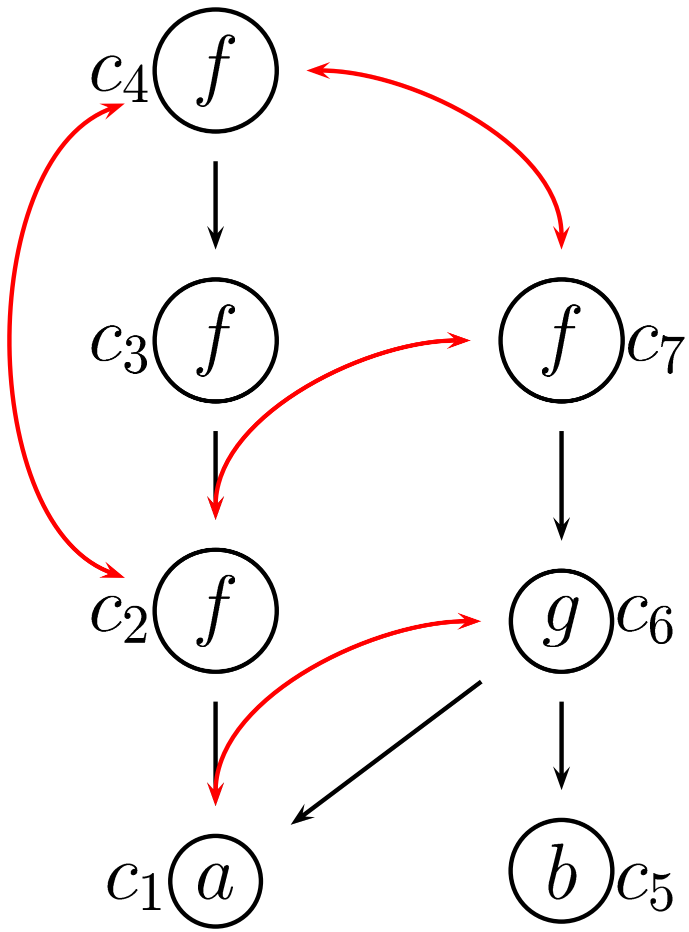
D-rules represent the term DAG:

$$
\begin{array}{rcl@{\qquad}rcl}
a & \to & c_1 & fc_1 & \to & c_2 \\
fc_2 & \to & c_3 & fc_3 & \to & c_4 \\
b & \to & c_5 & gc_1c_5 & \to & c_6 \\
fc_6 & \to & c_7 & & &
\end{array}
$$

C-rules represent an equivalence relation on vertices:

$$
c_1 = c_6 \qquad c_4 = c_7
$$

Illustration: D-rules and C-rules



D-rules represent the term DAG:

$$
\begin{array}{rcl@{\qquad}rcl}
a & \to & c_1 & fc_1 & \to & c_2 \\
fc_2 & \to & c_3 & fc_3 & \to & c_4 \\
b & \to & c_5 & gc_1c_5 & \to & c_6 \\
fc_6 & \to & c_7 & & &
\end{array}
$$

C-rules represent an equivalence relation on vertices:

$$c_1 = c_6 \qquad c_4 = c_7$$

Thus, $fffa \neq fa$,
i.e., $c_4 \neq c_2$ is a contradiction.

Abstract Congruence Closure

Formalizing the procedure:

U	:	set of new constants, denoted by c, d
K	:	Subset of U used until now
$\succ$	:	ordering on U
E, R	:	Finite sets of ground equations over $\Sigma \cup U$
(K, E, R)	:	State of derivation
$(\emptyset, E_0, \emptyset)$	:	Initial state

$$\text{Extension:}\ \frac{(K, E \cup \{s[f(c_1, \ldots, c_m)] = u\}, R)}{(K \cup \{c\}, E \cup \{s[c] = u\}, R \cup \{f(c_1, \ldots, c_m) \rightarrow c\})}$$

if $f \in \Sigma$, $c \in U - K$, and $c_1, \ldots, c_m \in K$.

Other Inference Rules

Simplification: $$\frac{(K, E \cup \{s[t] = u\}, R \cup \{t \to c\})}{(K, E \cup \{s[c] = u\}, R \cup \{t \to c\})}$$

Orientation: $$\frac{(K, E \cup \{c = d\}, R)}{(K, E, R \cup \{c \to d\})}$$ if $c \succ d$

Deletion: $$\frac{(K, E \cup \{t = t\}, R)}{(K, E, R)}$$

Other Inference Rules

Deduction: $$\frac{(K, E, R \cup \{t \to c,\ t \to d\})}{(K, E, R \cup \{c \to d,\ t \to d\})} \quad \text{if } c \succ d$$

Collapse: $$\frac{(K, E, R \cup \{s[c] \to d,\ c \to c'\})}{(K, E, R \cup \{s[c'] \to d,\ c \to c'\})} \quad \text{if } s[c] \not\equiv c$$

Composition: $$\frac{(K, E, R \cup \{t \to c,\ c \to d\})}{(K, E, R \cup \{t \to d,\ c \to d\})}$$

Definition

A ground rewrite system $R = D \cup C$ is an *(abstract) congruence closure* (over Σ and $K \subset U$) *for* E if

Definition

A ground rewrite system $R = D \cup C$ is an *(abstract) congruence closure* (over Σ and $K \subset U$) *for* E if

1. For every constant $c \in K$, there exists a term $t \in \mathcal{T}(\Sigma)$ s.t. $t \leftrightarrow_R^* c$,

Definition

A ground rewrite system $R = D \cup C$ is an *(abstract) congruence closure* (over Σ and $K \subset U$) *for* E if

1. For every constant $c \in K$, there exists a term $t \in \mathcal{T}(\Sigma)$ s.t. $t \leftrightarrow_R^* c$,
2. R is a terminating and confluent rewrite system, there is no infinite rewrite sequence using R, and whenever $s \leftrightarrow_R^* t$, it is also the case that

$$\begin{tikzcd} s \arrow[rr, leftrightarrow, dashed, "*"] \arrow[dr, "*"'] & & t \arrow[dl, "*"] \\ & \circ & \end{tikzcd}$$

Definition

A ground rewrite system $R = D \cup C$ is an *(abstract) congruence closure* (over Σ and $K \subset U$) *for* E if

1. For every constant $c \in K$, there exists a term $t \in \mathcal{T}(\Sigma)$ s.t. $t \leftrightarrow_R^* c$,
2. R is a terminating and confluent rewrite system,
3. E and R induce the same equational theory over Σ, i.e., for all terms $s, t \in \mathcal{T}(\Sigma)$, we have:

$$s \leftrightarrow_E^* t \text{ if and only if } s \leftrightarrow_R^* t.$$

Example: Abstract Closure

Let

$$E_0 = \{a = gab, \;\; fffa = fgab\}.$$

Then,

$$a = gab, \; fffa = fgab$$

Example: Abstract Closure

Let

$$E_0 = \{a = gab, \;\; fffa = fgab\}.$$

Then,

$$\frac{a = gab, \; fffa = fgab}{a \to c_1, \; fc_1 \to c_2, \; fc_2 \to c_3, \; fc_3 \to c_4, \; b \to c_5, \; gc_1c_5 \to c_6, \; fc_6 \to c_7, \; c_1 \to c_6, \; c_4 \to c_7}$$

Inference Rule Used: Extension, Simplification, Orientation

Example: Abstract Closure

Let

$$E_0 = \{a = gab, \;\; fffa = fgab\}.$$

Then,

$$\frac{\dfrac{a = gab, \;\; fffa = fgab}{a \to c_1, \;\; fc_1 \to c_2, \;\; fc_2 \to c_3, \;\; fc_3 \to c_4, \;\; b \to c_5, \;\; gc_1c_5 \to c_6, \;\; fc_6 \to c_7, \;\; c_1 \to c_6, \;\; c_4 \to c_7}}{fc_6 \to c_2}$$

Inference Rule Used: Collapse

Example: Abstract Closure

Let

$$E_0 = \{a = gab, \;\; fffa = fgab\}.$$

Then,

$$\frac{\dfrac{a = gab, \; fffa = fgab}{a \to c_1, \; fc_1 \to c_2, \; fc_2 \to c_3, \; fc_3 \to c_4, \; b \to c_5, \; gc_1c_5 \to c_6, \; fc_6 \to c_7, \; c_1 \to c_6, \; c_4 \to c_7}}{\dfrac{fc_6 \to c_2}{c_2 \to c_7}}$$

Inference Rule Used: Deduction

Example: Abstract Closure

Let

$$E_0 = \{a = gab, \;\; fffa = fgab\}.$$

Then,

$$a = gab, \; fffa = fgab$$

$$a \to c_1, \; fc_1 \to c_2, \; fc_2 \to c_3, \; fc_3 \to c_4, \; b \to c_5, \; gc_1c_5 \to c_6, \; fc_6 \to c_7, \; c_1 \to c_6, \; c_4 \to c_7$$

$$fc_6 \to c_2$$

$$c_2 \to c_7$$

The final abstract congruence closure

Correctness: Statement

If E is a finite set of equations of size n, then

1. Any derivation starting from $(\emptyset, E_0, \emptyset)$ reaches a saturated state $(K_\infty, \emptyset, R_\infty)$ in a finite number $T(n)$ of steps.
2. The set R_∞ is an abstract congruence closure for E_0.
3. $|K_\infty| = O(n)$ and if $c_1 \succ c_2 \succ \cdots \succ c_\delta$ is the longest chain induced by $\succ$ over K_∞, then $T(n) = O(n\delta)$.
4. Using a standard trick, δ can be bounded by $O(\log(n))$.
5. For special cases, $\delta = O(1)$.

Size of E is the length of string representing E

$\succ$ is required to successfully orient all generated equations

Correctness: Soundness/Completeness

- Extension: If $(K \cup \{c\}, E', R')$ is obtained from (K, E, R) using Extension, then

$$\forall s, t \in \mathcal{T}(\Sigma \cup K). \quad s \leftrightarrow^*_{E,R} t \quad \text{iff} \quad s \leftrightarrow^*_{E',R'} t.$$

- All other rules are *standard Knuth-Bendix completion* rules.
- Equations in E can always be removed by extension, orientation, or deletion.
- Every rewrite rule in R is decreasing in a suitable reduction ordering.
- By correctness of completion, R_∞ is convergent.

Correctness Proof: Complexity

- Number of Σ-symbols in E never increases and Extension always decreases it.
  $\therefore |K_\infty| \leq n$.
- $|E \cup R|$ is increased by Extension alone.
  $\therefore$ after all Extension steps, $|E \cup R| = O(n)$.
  $\therefore |E_\infty| = O(n)$.
- Consider a rewrite rule in E.

$$f(c_1, \ldots, c_m) \ \rightarrow \ c$$

  Superposition, Collapse, Composition inference rules simplify one of $c_1, c_2, \ldots, c_m, c$, or rewrite the LHS.

Correctness Proof: Complexity

- Number of Σ-symbols in E never increases and Extension always decreases it.
  $\therefore |K_\infty| \leq n$.
- $|E \cup R|$ is increased by Extension alone.
  $\therefore$ after all Extension steps, $|E \cup R| = O(n)$.
  $\therefore |E_\infty| = O(n)$.
- Consider a rewrite rule in E.

$$\underline{f}(\underline{c_1}, \ldots, \underline{c_m}) \;\rightarrow\; \underline{c}$$

If k inferences are applied at each position, then

$$c_1 \succ c_2 \succ c_3 \succ \ldots \succ c_{k-1} \qquad\qquad \therefore k < \delta$$

Correctness Proof: Complexity

- Number of Σ-symbols in E never increases and Extension always decreases it.
  $\therefore |K_\infty| \leq n$.
- $|E \cup R|$ is increased by Extension alone.
  $\therefore$ after all Extension steps, $|E \cup R| = O(n)$.
  $\therefore |E_\infty| = O(n)$.
- Consider a rewrite rule in E.

$$\underline{f}(\underline{c_1}, \ldots, \underline{c_m}) \;\rightarrow\; \underline{c}$$

This rule contributes at most $(m+2)\delta$ inferences.

Correctness Proof: Complexity

- Number of Σ-symbols in E never increases and Extension always decreases it.
  $\therefore |K_\infty| \leq n$.
- $|E \cup R|$ is increased by Extension alone.
  $\therefore$ after all Extension steps, $|E \cup R| = O(n)$.
  $\therefore |E_\infty| = O(n)$.
- The set $|E|$ can contribute at most $n\delta$ Superposition, Collapse, and Composition inferences.

Correctness Proof: Complexity

- Number of Σ-symbols in E never increases and Extension always decreases it.
  $\therefore |K_\infty| \leq n$.
- $|E \cup R|$ is increased by Extension alone.
  $\therefore$ after all Extension steps, $|E \cup R| = O(n)$.
  $\therefore |E_\infty| = O(n)$.
- The set $|E|$ can contribute at most $n\delta$ Superposition, Collapse, and Composition inferences.
- Number of Extension, Simplification, Orientation, and Deletion steps is $O(n)$.
  $\therefore$ derivation length = $O(n\delta) = O(n^2)$.

Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.

$$c_6$$

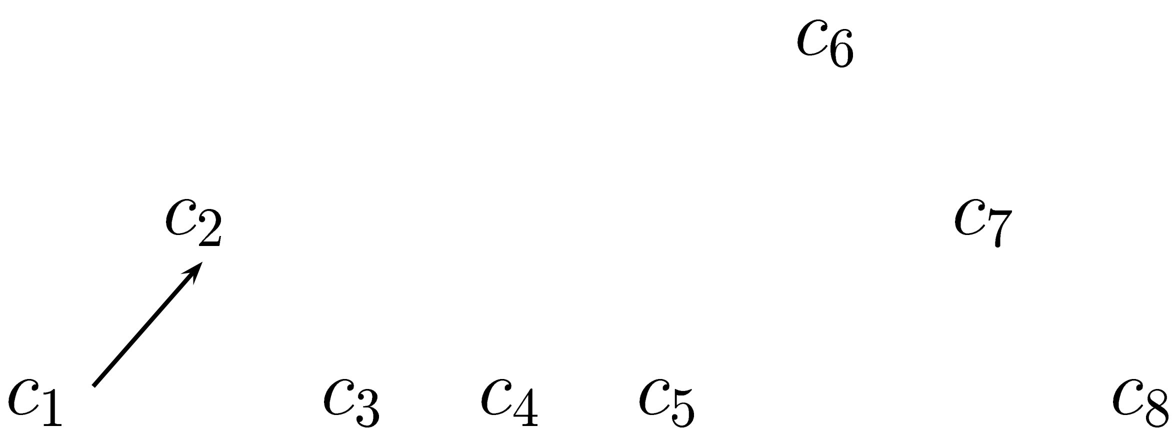
$$c_2 \qquad\qquad\qquad\qquad\qquad\qquad c_7$$

$$c_1 \qquad c_3 \quad c_4 \quad c_5 \qquad\qquad\qquad c_8$$

Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:
  $c_1 = c_2$,

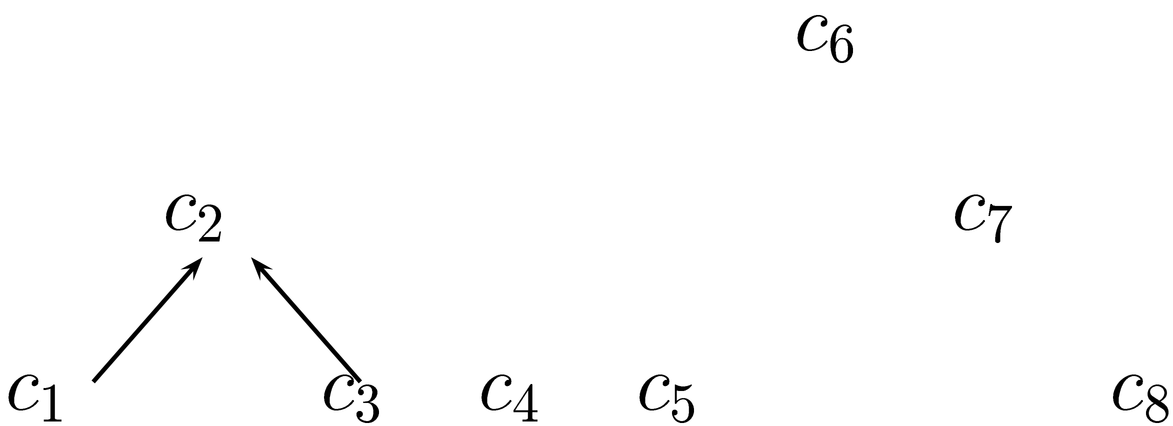


Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:

  $c_1 = c_2,\ \ c_2 = c_3,$

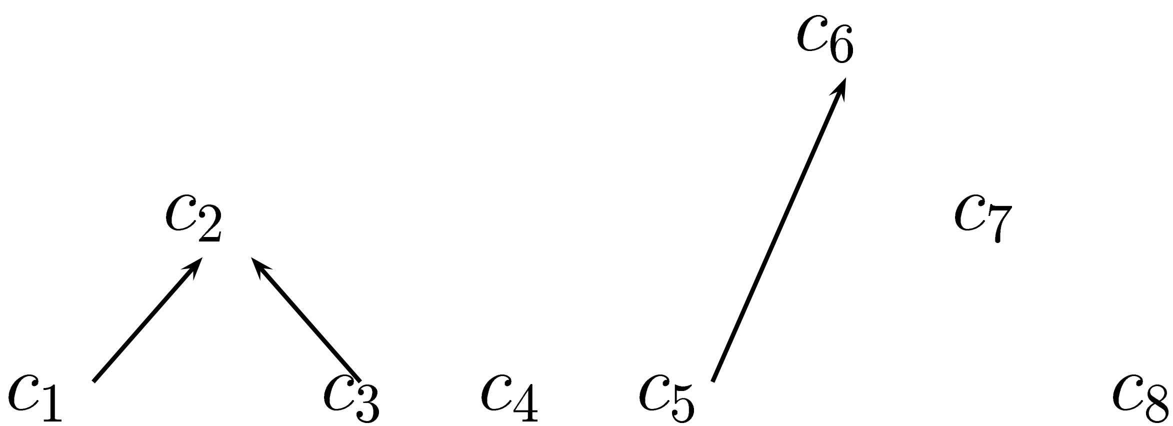


Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:

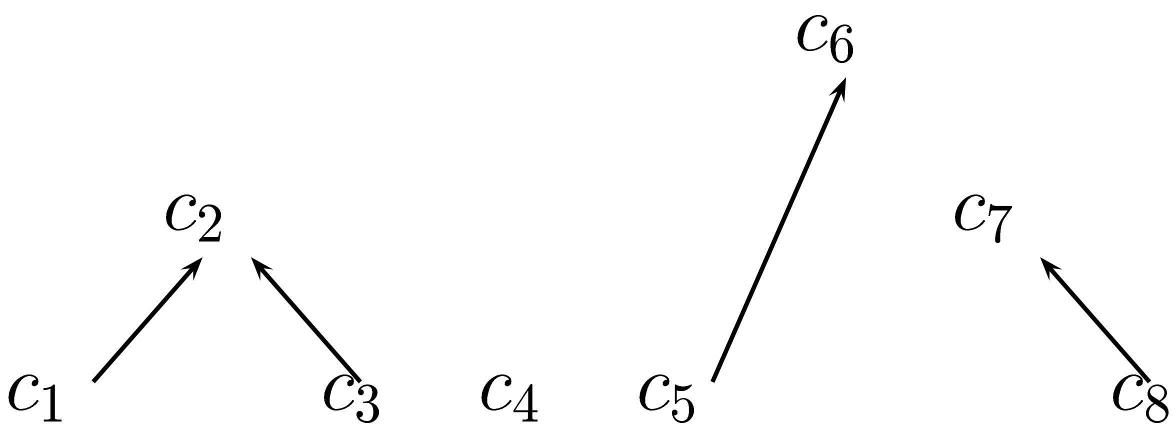
  $c_1 = c_2,\ \ c_2 = c_3,\ \ c_5 = c_6,$



Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:
  $c_1 = c_2,\ c_2 = c_3,\ c_5 = c_6,\ c_7 = c_8,$

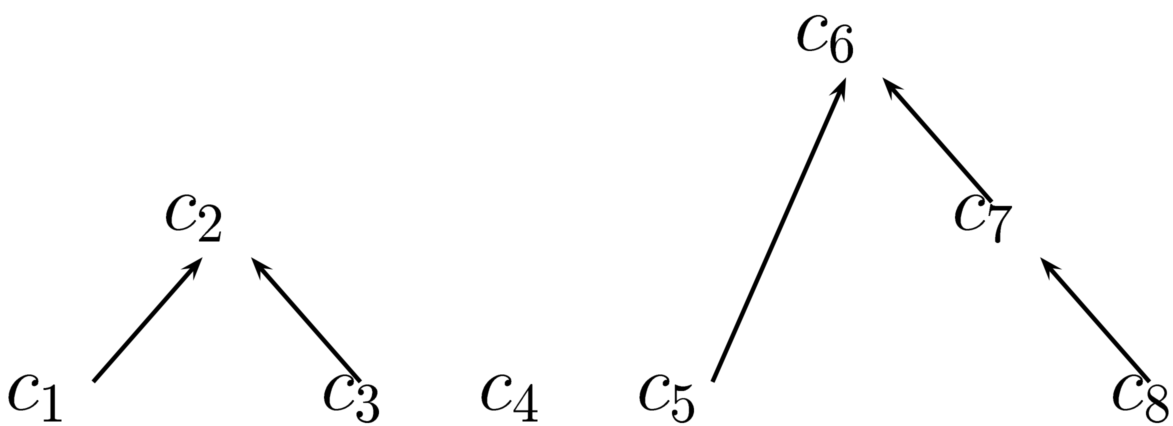


Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:

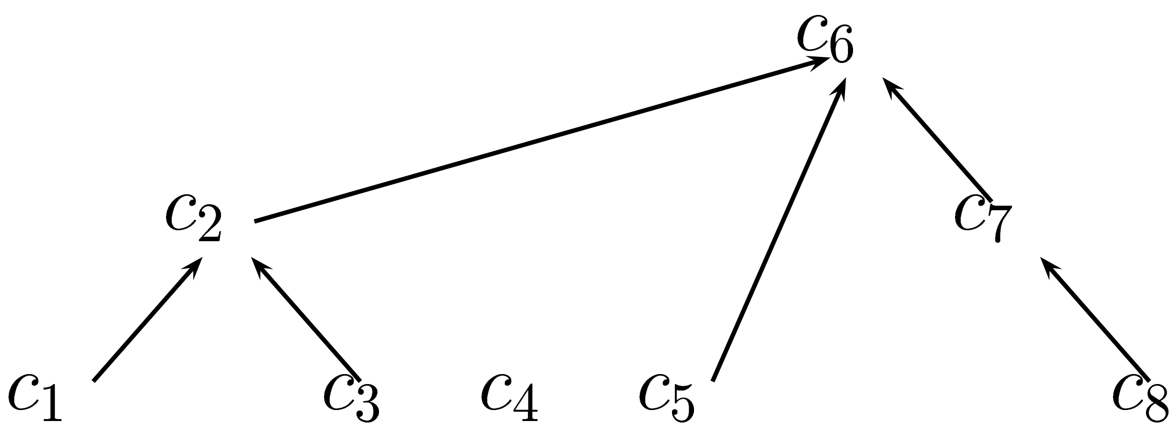
  $c_1 = c_2, \;\; c_2 = c_3, \;\; c_5 = c_6, \;\; c_7 = c_8, \;\; c_6 = c_7,$



Efficient Variants

Choosing $\succ$ at run-time so that δ is small:

- Consider the set $K = \{c_1, \ldots, c_8\}$ of eight constants.
- Say we generate equations, which need to be oriented, in the following order:
  $c_1 = c_2,\ c_2 = c_3,\ c_5 = c_6,\ c_7 = c_8,\ c_6 = c_7,\ c_2 = c_6.$



Therefore, $\delta = O(\log(n))$.

Specialized Algorithms

- Shostak's dynamic congruence closure:

$$\mathbf{Shos} \;=\; [(\mathbf{Sim}^*{\cdot}\mathbf{Ext}^*)^* \cdot (\mathbf{Del}{\cup}\mathbf{Ori}) \cdot (\mathbf{Col}{\cdot}\mathbf{Ded}^*)^*]^*$$

- Downey-Sethi-Tarjan's Algorithm: uses the $O(n\log(n))$ trick with

$$\mathbf{DST} \;=\; [(\mathbf{Col}{\cdot}(\mathbf{Ded}{\cup}\{\epsilon\}))^* \cdot (\mathbf{Sim}^*{\cdot}(\mathbf{Del}{\cup}\mathbf{Ori}))^*]^*$$

- Nelson and Oppen's Algorithm:

$$\mathbf{NO} \;=\; [(\mathbf{Sim}^* \;\cdot\; (\mathbf{Ori} \cup \mathbf{Del}) \;\cdot\; \mathbf{NODed}^*]^*$$

where NODed rule corresponds to superposition modulo C.

Example: Shostak's CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$a = gab,\ fffa = fgab$$

Example: Shostak's CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{a = gab,\ fffa = fgab}{a \to c_1,\ b \to c_5,\ gc_1c_5 \to c_1}$$

Example: Shostak's CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$a = gab,\ fffa = fgab$

$a \to c_1,\ b \to c_5,\ gc_1c_5 \to c_1$

$fffc_1 = fc_1$

Example: Shostak's CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{\dfrac{\dfrac{a = gab,\ fffa = fgab}{a \to c_1,\ b \to c_5,\ gc_1c_5 \to c_1}}{fffc_1 = fc_1}}{fc_1 \to c_2,\ fc_2 \to c_3,\ fc_3 \to c_2}$$

Example: Shostak's CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{\dfrac{\dfrac{a = gab,\ fffa = fgab}{a \to c_1,\ b \to c_5,\ gc_1c_5 \to c_1}}{fffc_1 = fc_1}}{fc_1 \to c_2,\ fc_2 \to c_3,\ fc_3 \to c_2}$$

Final congruence closure

$$\{a \to c_1, b \to c_5, gc_1c_5 \to c_1, fc_1 \to c_2, fc_2 \to c_3, fc_3 \to c_2\}$$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$a = gab, \; fffa = fgab$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{a = gab, \;\; fffa = fgab}{a \to c_1, \;\; \ldots, fc_6 \to c_7, \;\; c_1 = c_6, \;\; c_4 = c_7}$$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{\dfrac{a = gab,\ fffa = fgab}{a \to c_1,\ \ldots, fc_6 \to c_7, c_1 = c_6,\ c_4 = c_7}}{c_1 \to c_6,\ fc_6 \to c_2,\ c_2 = c_7,\ gc_6c_5 \to c_6}$$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$a = gab,\ fffa = fgab$$

$$a \to c_1,\ \ldots, fc_6 \to c_7, c_1 = c_6,\ c_4 = c_7$$

$$c_1 \to c_6,\ fc_6 \to c_2,\ c_2 = c_7,\ gc_6c_5 \to c_6$$

$$c_4 \to c_7$$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$a = gab,\ fffa = fgab$$

$$a \to c_1,\ \ldots,\ fc_6 \to c_7, c_1 = c_6,\ c_4 = c_7$$

$$c_1 \to c_6,\ fc_6 \to c_2,\ c_2 = c_7,\ gc_6c_5 \to c_6$$

$$c_4 \to c_7$$

$$c_2 \to c_7,\ fc_7 \to c_3$$

Example: DST CC

$$E_0 = \{a = gab, fffa = fgab\}$$

$$\frac{\frac{\frac{\frac{a = gab,\ fffa = fgab}{a \to c_1,\ \ldots,\ fc_6 \to c_7,\ c_1 = c_6,\ c_4 = c_7}}{c_1 \to c_6,\ fc_6 \to c_2,\ c_2 = c_7,\ gc_6c_5 \to c_6}}{c_4 \to c_7}}{c_2 \to c_7,\ fc_7 \to c_3}$$

Final congruence closure is:

$\{a \to c_1, fc_7 \to c_3, fc_3 \to c_4, b \to c_5, fc_6c_5 \to c_6, fc_6 \to c_7, c_1 \to c_6, c_2 \to c_7, c_4 \to c_7\}$

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Combination of Theories

$$\begin{array}{lcl} \Sigma & = & \Sigma_1 \cup \Sigma_2 \\ \mathcal{T}_1, \mathcal{T}_2 & : & \text{Theories over } \Sigma_1 \text{ and } \Sigma_2 \\ \mathcal{T} & = & \text{Deductive closure of } \mathcal{T}_1 \cup \mathcal{T}_2 \end{array}$$

Problem1. Is $\mathcal{T}$ consistent?

Problem2. Given satisfiability procedures for (quantifier-free) conjunction of literals in $\mathcal{T}_1$ and $\mathcal{T}_2$, how to decide satisfiability in $\mathcal{T}$?

Problem3. What is the complexity of the combination procedure?

Stably-Infinite Theories

A theory is *stably-infinite* if every satisfiable QFF is satisfiable in an infinite model.

Example. Theories with only finite models are not stably infinite. Thus, theory induced by the axiom $\forall x, y, z.(x = y \vee y = z \vee z = x)$ is not stably-infinite.

Proposition. If E is an equational theory, then $E \cup \{\exists x, y.x \neq y\}$ is stably-infinite.
Proof. If M is a model, then $M \times M$ is a model as well. Hence, by compactness, there is an infinite model.

Proposition. The union of two consistent, disjoint, stably-infinite theories is consistent.
Proof. Later!

Convexity

A theory is *convex* if whenever a conjunction of literals implies a disjunction of atomic formulas, it also implies one of the disjuncts.

Example. The theory of integers over a signature containing $<$ is not convex. The formula $1 < x \ \wedge \ x < 4$ implies $x = 2 \ \vee \ x = 3$, but it does not imply either $x = 2$ or $x = 3$ independently.

Example. The theory of rationals over the signature $\{+, <\}$ is convex.

Example. Equational theories are convex, but need not be stably-infinite.

Convexity: Observation

Proposition. A convex theory $\mathcal{T}$ with no trivial models is stably-infinite.
Proof. If not, then for some QFF ϕ, $\mathcal{T} \cup \phi$ has only finite models. Thus, ϕ implies a disjunction $\vee_{i,j} x_i = x_j$, without implying any disjunct.

Example. If E is an equational theory, then $E \cup \{\exists x, y. x \neq y\}$ has no trivial models, and hence it is stably-infinite.

Nelson-Oppen Combination Result

Theorem 1 *Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be consistent, stably-infinite theories over disjoint (countable) signatures. Assume satisfiability of (quantifier-free) conjunction of literals can be decided in $O(T_1(n))$ and $O(T_2(n))$ time respectively. Then,*

1. *The combined theory $\mathcal{T}$ is consistent and stably infinite.*
2. *Satisfiability of (quantifier-free) conjunction of literals in $\mathcal{T}$ can be decided in $O(2^{n^2} * (T_1(n) + T_2(n)))$ time.*
3. *If $\mathcal{T}_1$ and $\mathcal{T}_2$ are convex, then so is $\mathcal{T}$ and satisfiability in $\mathcal{T}$ is in $O(n^4 * (T_1(n) + T_2(n)))$ time.*

Proof. Later.

Examples

Convexity is important for point (3) above.

	$\mathcal{T}_1$	$\mathcal{T}_2$	$\mathcal{T}_1 \cup \mathcal{T}_2$
Signature	Σ_F	$\{\mathbb{Z}, <\}$	$\{\mathbb{Z}, <\} \cup \Sigma_F$
Satisfiability	$O(n \log(n))$	$O(n^2)$	NP-complete!

Note that $\mathcal{T}_2$ is not convex.

We can allow a "add constant" operator in signature of $\mathcal{T}_2$. Atomic formulae are of the form $x - y < c$, for some constant c, and satisfiability can be tested by searching for negative cycles in a "difference graph".

For NP-completeness of the union theory, see [Pratt77].

Nelson-Oppen Result: Correctness

Recall the theorem. The combination procedure:

Initial State : ϕ is a conjunction of literals over $\Sigma_1 \cup \Sigma_2$.

Purification : Preserving satisfiability, transform ϕ to $\phi_1 \wedge \phi_2$, s.t. ϕ_i is over Σ_i.

Interaction : Guess a partition of $\mathcal{V}(\phi_1) \cap \mathcal{V}(\phi_2)$ into disjoint subsets.
Express it as a conjunction of literals ψ.
Example. The partition $\{x_1\}, \{x_2, x_3\}$ is represented as $x_2 = x_3 \ \wedge \ x_1 \neq x_2 \ \wedge \ x_1 \neq x_3$.

Component Procedures : Use individual procedures to decide if $\phi_i \ \wedge \ \psi$ is satisfiable.

Return : If both answer yes, return yes. No, otherwise.

Separating Concerns: Purification

Purification: $$\frac{\phi \;\wedge\; P(\ldots, s[t], \ldots)}{\phi \;\wedge\; P(\ldots, s[x], \ldots) \;\wedge\; t = x}$$ if $s[t]$ is not a variable.

Proposition. Purification is satisfiability preserving: if ϕ' is obtained from ϕ by purification, then ϕ is satisfiable in the union theory iff ϕ' is satisfiable in the union theory.

Proposition. Purification is terminating.

Proposition. Exhaustive application results in conjunction where each conjunct is over exactly one signature.

Purification: Illustration

$$f(\underbrace{x-1}_{u_1}) - 1 \;=\; x + 1,\; f(y) + 1 \;=\; y - 1,\; y + 1 \;=\; x$$

Purification: Illustration

$$\frac{f(\underbrace{x-1}_{u_1})-1 \;=\; x+1,\; f(y)+1 \;=\; y-1,\; y+1 \;=\; x}{\underbrace{f(u)}_{u_2}-1 \;=\; x+1,\; f(y)+1 \;=\; y-1,\; y+1 \;=\; x}$$

$$x - 1 = u_1$$

Purification: Illustration

$$f(\underbrace{x-1}_{u_1})-1 = x+1,\ f(y)+1 = y-1,\ y+1 = x$$

$$\underbrace{f(u)}_{u_2}-1 = x+1,\ f(y)+1 = y-1,\ y+1 = x$$

$$u_2-1 = x+1,\ \underbrace{f(y)}_{u_3}+1 = y-1,\ y+1 = x$$

$x-1=u_1$,$f(u)=u_2$

Purification: Illustration

$$f(\underbrace{x-1}_{u_1}) - 1 \;=\; x+1,\; f(y)+1 \;=\; y-1,\; y+1 \;=\; x$$

$$\underbrace{f(u)}_{u_2} - 1 \;=\; x+1,\; f(y)+1 \;=\; y-1,\; y+1 \;=\; x$$

$$u_2 - 1 \;=\; x+1,\; \underbrace{f(y)}_{u_3}+1 \;=\; y-1,\; y+1 \;=\; x$$

$$u_2 - 1 \;=\; x+1,\; u_3+1 \;=\; y-1,\; y+1 \;=\; x$$

$$x-1=u_1\;, f(u)=u_2\;, f(y)=u_3$$

NO Procedure Soundness

Each step is satisfiability preserving.
Say ϕ is satisfiable (in the combination).

NO Procedure Soundness

Each step is satisfiability preserving.
Say ϕ is satisfiable (in the combination).

1. *Purification:* $\therefore\ \phi_1 \ \wedge\ \phi_2$ is satisfiable.

NO Procedure Soundness

Each step is satisfiability preserving.
Say ϕ is satisfiable (in the combination).

1. *Purification:* $\therefore\ \phi_1 \ \wedge \ \phi_2$ is satisfiable.
2. *Interaction:* $\therefore$ for some partition ψ, $\phi_1 \ \wedge \ \phi_2 \ \wedge \ \psi$ is satisfiable.

NO Procedure Soundness

Each step is satisfiability preserving.
Say ϕ is satisfiable (in the combination).

1. *Purification:* $\therefore\ \phi_1 \wedge \phi_2$ is satisfiable.
2. *Interaction:* $\therefore$ for some partition ψ, $\phi_1 \wedge \phi_2 \wedge \psi$ is satisfiable.
3. *Components Procedures:* $\therefore$, $\phi_1 \wedge \psi$ and $\phi_2 \wedge \psi$ are both satisfiable in component theories.

Therefore, if the procedure returns unsatisfiable, then the formula ϕ is indeed unsatisfiable.

NO Procedure Correctness

Suppose the procedure returns satisfiable.

NO Procedure Correctness

Suppose the procedure returns satisfiable.

- Let ψ be the partition and $\mathbb{A}$ and $\mathbb{B}$ be models of $\mathcal{T}_1 \ \wedge \ \phi_1 \ \wedge \ \psi$ and $\mathcal{T}_2 \ \wedge \ \phi_2 \ \wedge \ \psi$.

NO Procedure Correctness

Suppose the procedure returns satisfiable.

- Let ψ be the partition and $\mathbb{A}$ and $\mathbb{B}$ be models of $\mathcal{T}_1 \ \wedge \ \phi_1 \ \wedge \ \psi$ and $\mathcal{T}_2 \ \wedge \ \phi_2 \ \wedge \ \psi$.
- Component theories are stably-infinite, $\therefore$ assume models are infinite (of same cardinality).
- Let h be a bijection between A and B s.t. $h(x^{\mathbb{A}}) = x^{\mathbb{B}}$ for each shared variable x. We can do this $\because$ of ψ.

NO Procedure Correctness

Suppose the procedure returns satisfiable.

- Let ψ be the partition and $\mathbb{A}$ and $\mathbb{B}$ be models of $\mathcal{T}_1 \ \wedge \ \phi_1 \ \wedge \ \psi$ and $\mathcal{T}_2 \ \wedge \ \phi_2 \ \wedge \ \psi$.
- Component theories are stably-infinite, $\therefore$ assume models are infinite (of same cardinality).
- Let h be a bijection between A and B s.t. $h(x^{\mathbb{A}}) = x^{\mathbb{B}}$ for each shared variable x. We can do this $\because$ of ψ.
- Extend $\mathbb{B}$ to $\overline{\mathbb{B}}$ by interpretations of symbols in Σ_1:

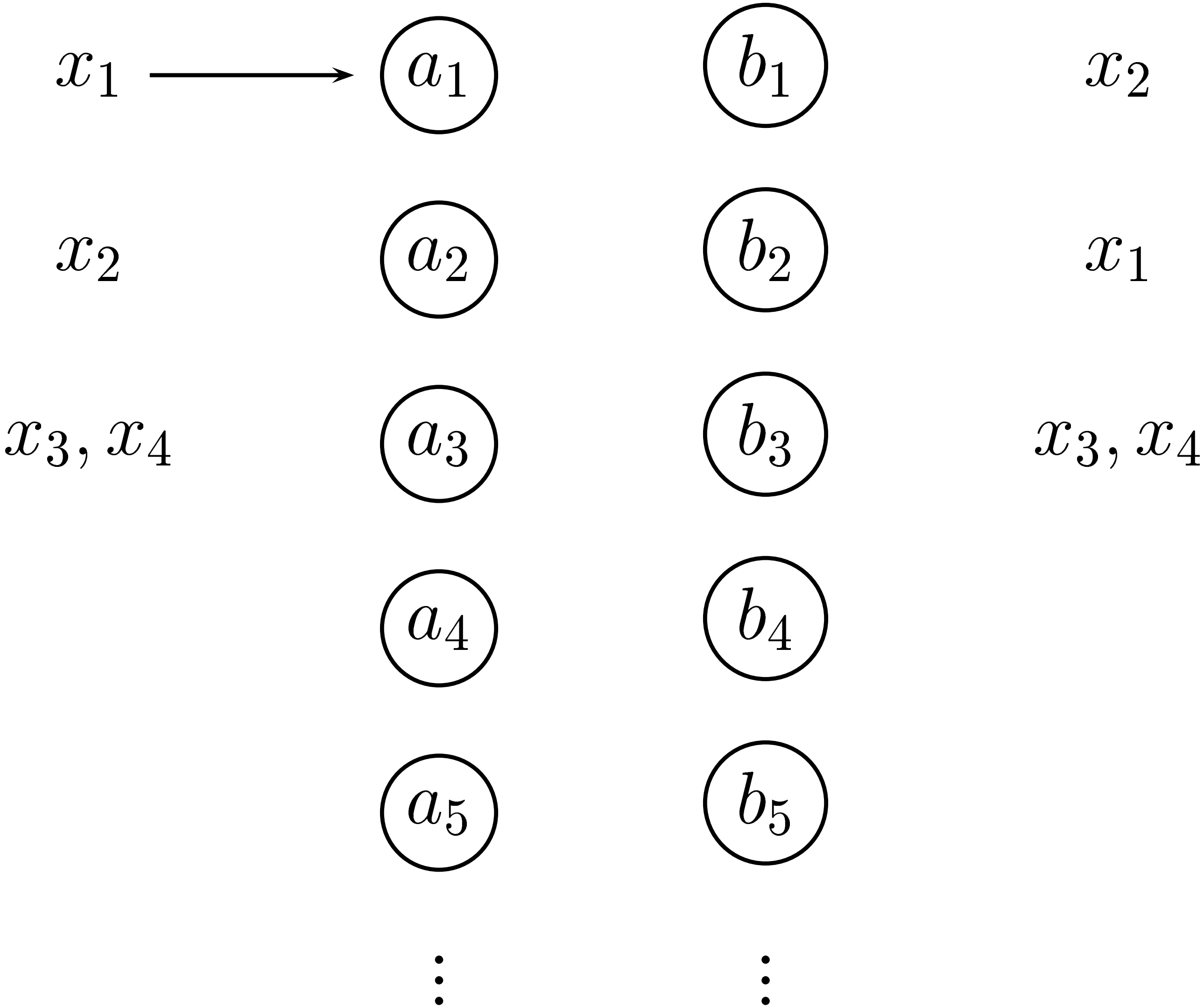
$$f^{\overline{\mathbb{B}}}(b_1, \ldots, b_k) = h(f^{\mathbb{A}}(h^{-1}(b_1), \ldots, h^{-1}(b_k)))$$

Such an extended $\overline{\mathbb{B}}$ is a model of

$$\mathcal{T}_1 \ \wedge \ \mathcal{T}_2 \ \wedge \phi_1 \ \wedge \ \phi_2 \ \wedge \ \psi$$

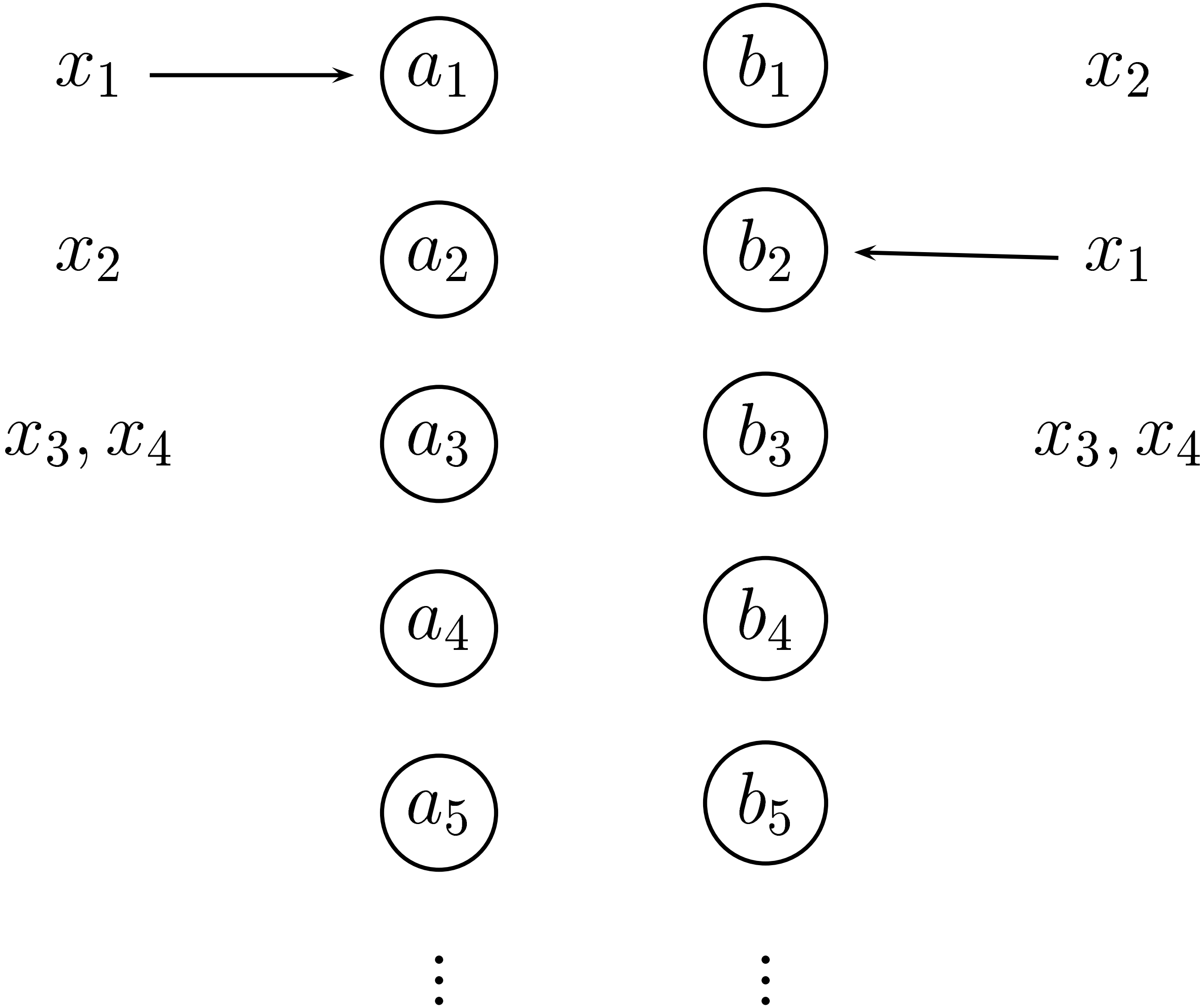
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



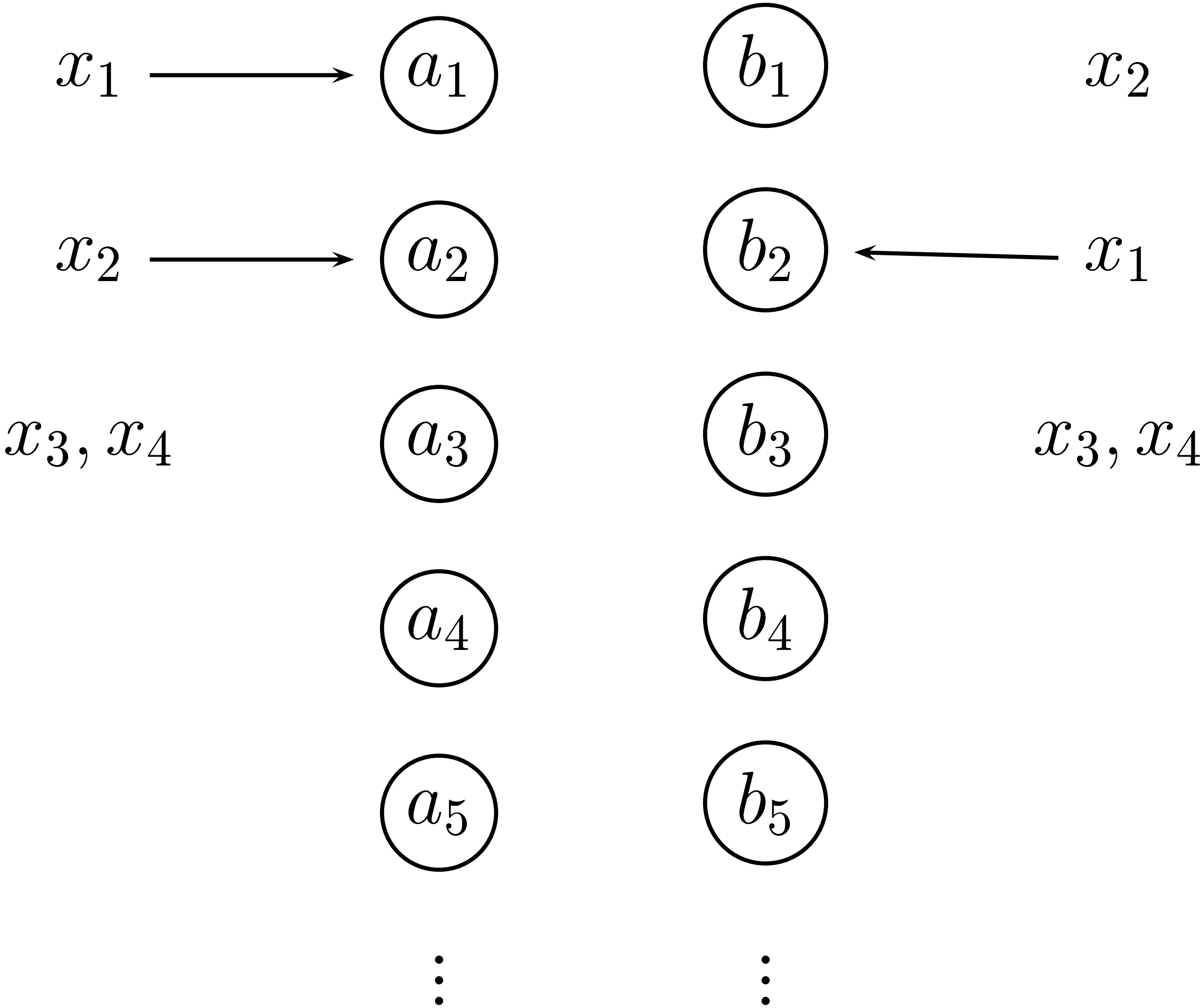
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



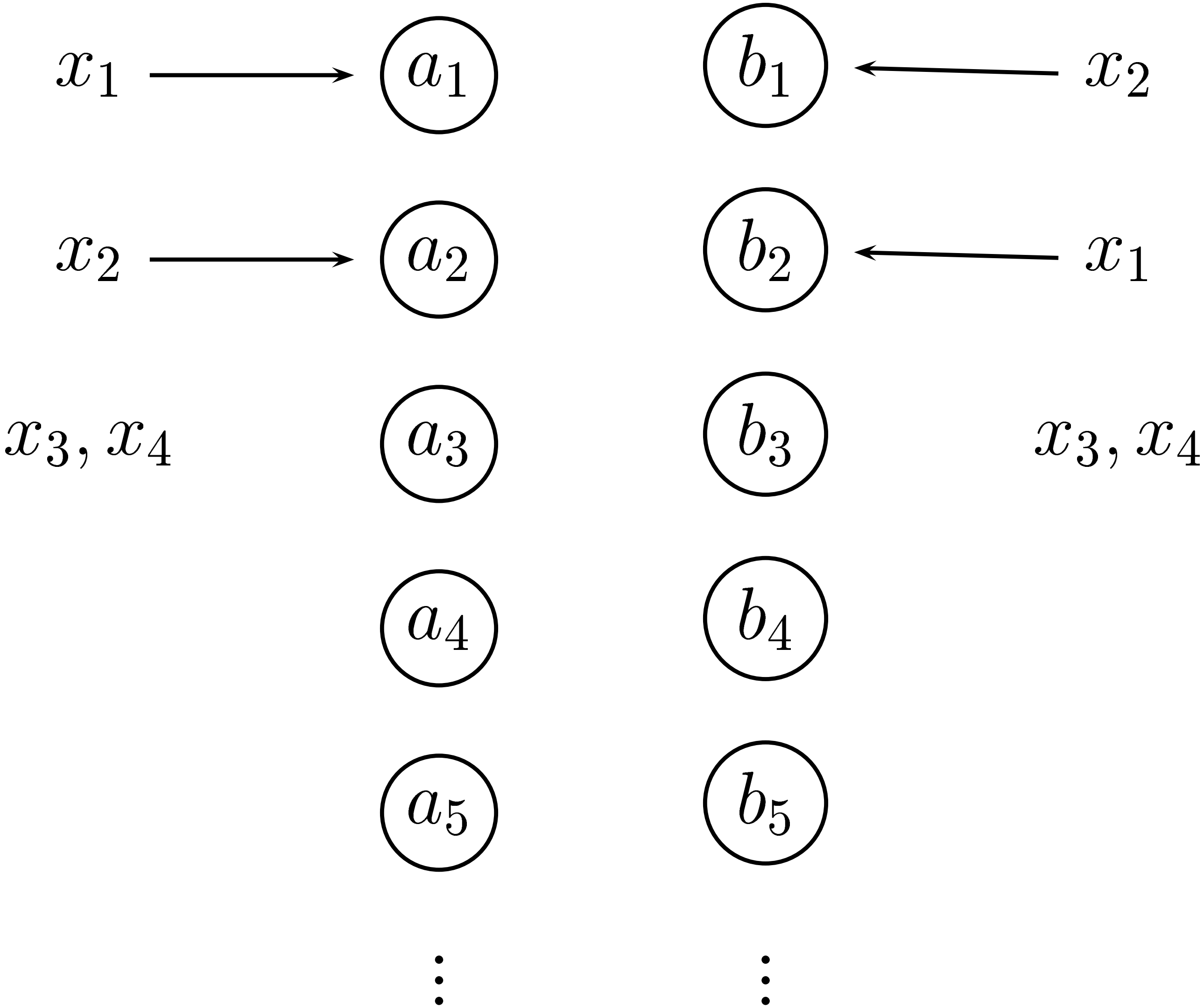
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



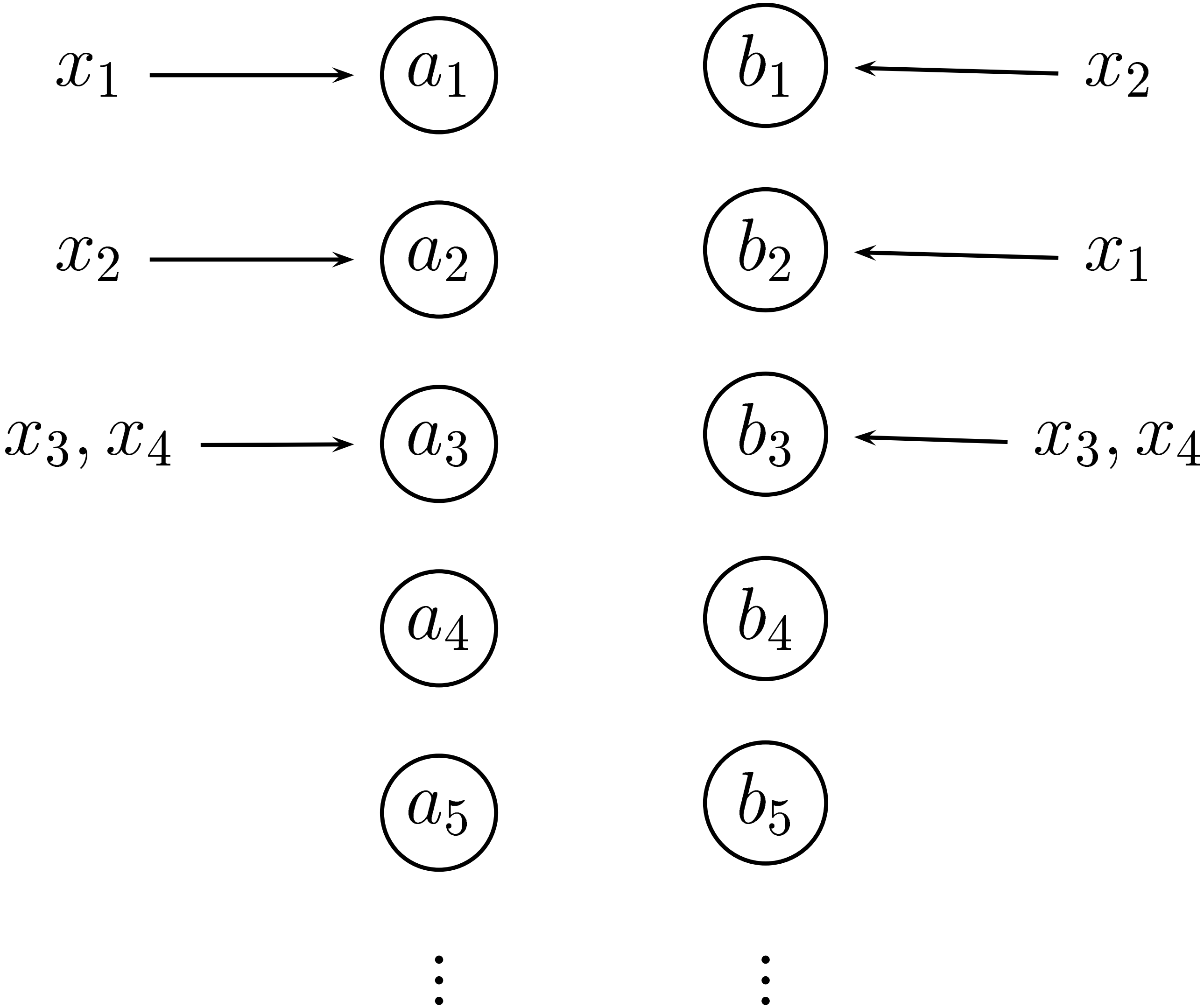
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



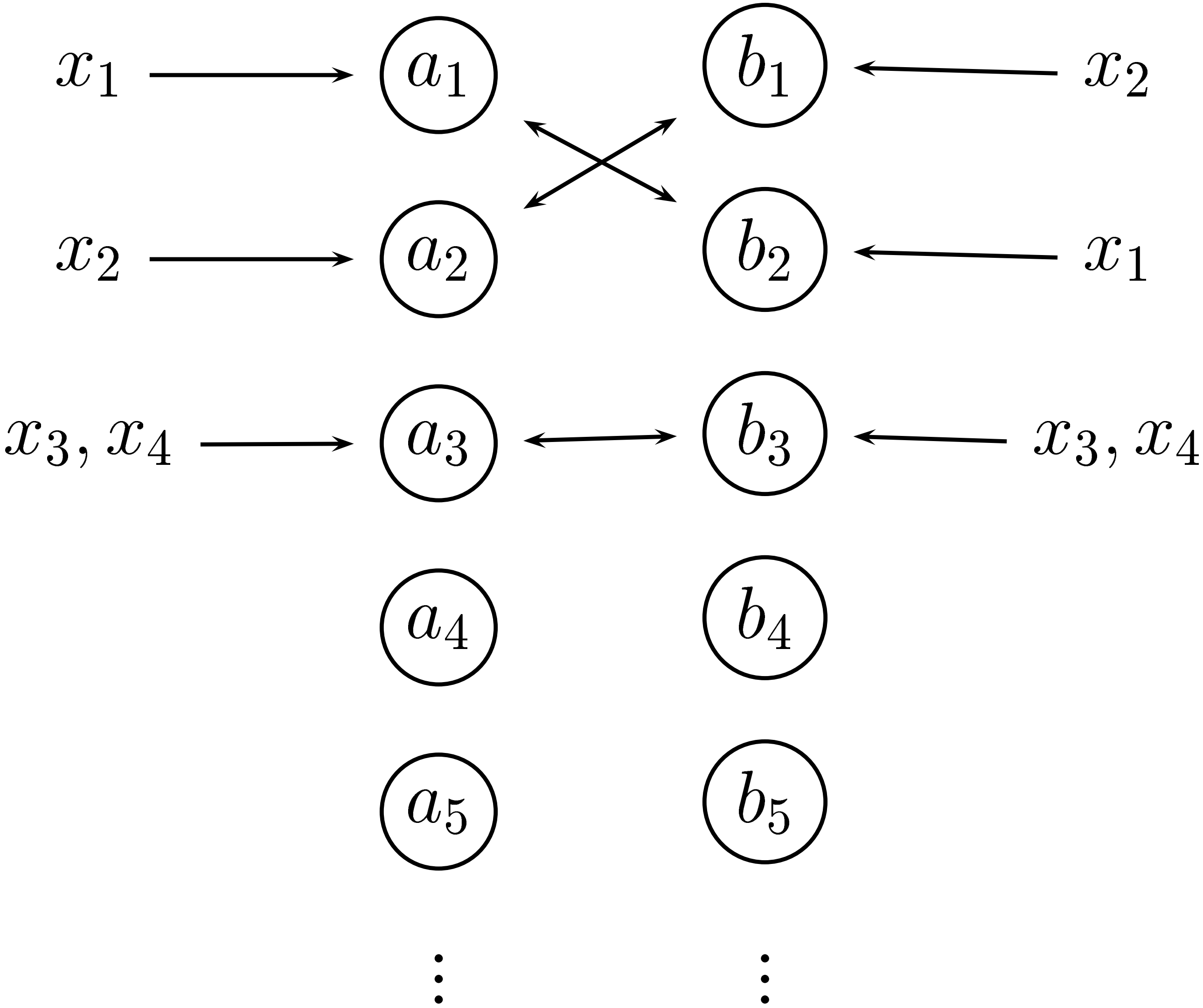
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



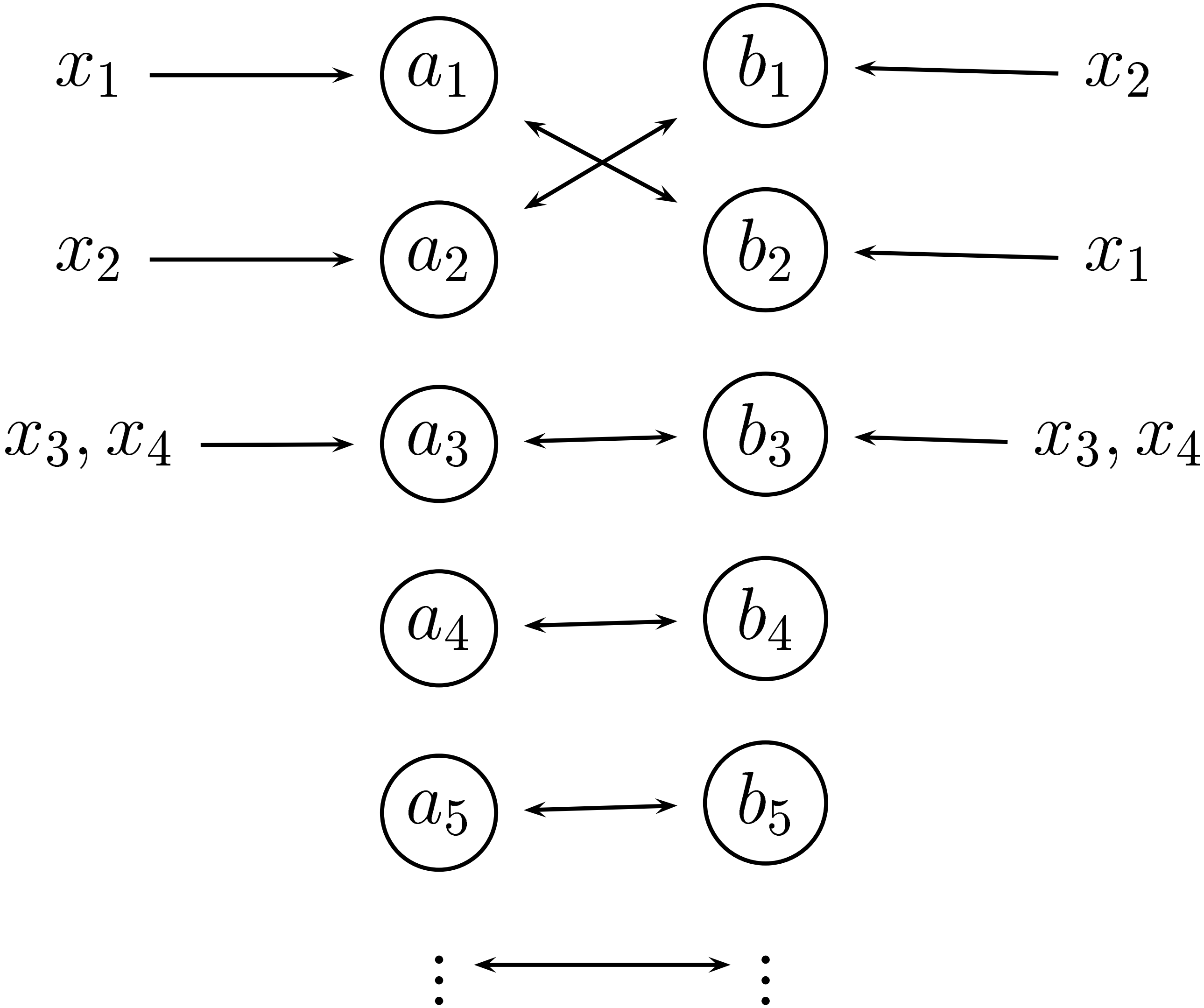
Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



Model Construction Picture

Consider $\mathcal{T}_i$-models $\mathbb{A}$ and $\mathbb{B}$ of $\phi_i \wedge \psi$:



NO Procedure Complexity

Proposition. The non-deterministic procedure can be determinised to give a $O(2^{n^2} * (T_1(n) + T_2(n)))$-time algorithm.

Proof.

NO Procedure Complexity

Proposition. The non-deterministic procedure can be determinised to give a $O(2^{n^2} * (T_1(n) + T_2(n)))$-time algorithm.

Proof.

1. Number of purification steps $< n$ and size of resulting $\phi_1 \ \wedge \ \phi_2$ is $O(n)$.

NO Procedure Complexity

Proposition. The non-deterministic procedure can be determinised to give a $O(2^{n^2} * (T_1(n) + T_2(n)))$-time algorithm.

Proof.

1. Number of purification steps $< n$ and size of resulting $\phi_1 \ \wedge \ \phi_2$ is $O(n)$.
2. Number of partition of a set with n variables: $B(n) < 2^{n^2}$.

NO Procedure Complexity

Proposition. The non-deterministic procedure can be determinised to give a $O(2^{n^2} * (T_1(n) + T_2(n)))$-time algorithm.

Proof.

1. Number of purification steps $< n$ and size of resulting $\phi_1 \ \wedge \ \phi_2$ is $O(n)$.
2. Number of partition of a set with n variables: $B(n) < 2^{n^2}$.
3. For each $B(n)$ choices, the component procedures take $T_1(n)$ and $T_2(n)$-time respectively.

NO Deterministic Procedure

Instead of guessing, we can deduce the equalities to be shared. The new combination procedure:

Purification : As before.

Interaction : Deduce an equality $x = y$:

$$\mathcal{T}_1 \vdash (\phi_1 \Rightarrow x = y)$$

Update $\phi_2 \;:=\; \phi_2 \;\wedge\; x = y$. And vice-versa. Repeat until no further changes to get ϕ_{i_∞}.

Component Procedures : Use individual procedures to decide if ϕ_{i_∞} is satisfiable.

Note, $\mathcal{T}_i \vdash (\phi_i \Rightarrow x = y)$ iff $\phi_1 \wedge x = y$ is not satisfiable in $\mathcal{T}_i$.

Deterministic Version: Correctness

Each step is satisfiability preserving, $\therefore$ soundness follows.

Deterministic Version: Correctness

Each step is satisfiability preserving, $\therefore$ soundness follows.

Assume that the theories are convex.

- Let ϕ_{i_∞} be satisfiable.

Deterministic Version: Correctness

Each step is satisfiability preserving, $\therefore$ soundness follows.

Assume that the theories are convex.

- Let ϕ_{i_∞} be satisfiable.
- If $\{x_1, \ldots, x_m\}$ is the set of variables not yet identified, $\mathcal{T}_i \not\vdash \phi_{i_\infty} \Rightarrow (x_j = x_k)$.

Deterministic Version: Correctness

Each step is satisfiability preserving, $\therefore$ soundness follows.

Assume that the theories are convex.

- Let ϕ_{i_∞} be satisfiable.
- If $\{x_1, \ldots, x_m\}$ is the set of variables not yet identified, $\mathcal{T}_i \not\vdash \phi_{i_\infty} \Rightarrow (x_j = x_k)$.
- By convexity, $\mathcal{T}_i \not\vdash \phi_{i_\infty} \Rightarrow \bigvee_{j \neq k}(x_j = x_k)$.

Deterministic Version: Correctness

Each step is satisfiability preserving, $\therefore$ soundness follows.

Assume that the theories are convex.

- Let ϕ_{i_∞} be satisfiable.
- If $\{x_1, \ldots, x_m\}$ is the set of variables not yet identified, $\mathcal{T}_i \not\vdash \phi_{i_\infty} \Rightarrow (x_j = x_k)$.
- By convexity, $\mathcal{T}_i \not\vdash \phi_{i_\infty} \Rightarrow \bigvee_{j \neq k}(x_j = x_k)$.
- $\therefore \phi_{i_\infty} \wedge \bigwedge_{j \neq k}(x_j \neq x_k)$ is satisfiable.
- The proof is now identical to the previous case.

Deterministic Version: Complexity

For convex theories, the combination procedure runs in $O(n^4 * (T_1(n) + T_2(n)))$ time:

1. Identifying if an equality $x = y$ is implied by ϕ_i takes $O(n^2 * T_i(n))$ time.
2. Since there are $O(n^2)$ possible equalities between variables, fixpoint is reached in $O(n^2)$ iterations.

Modularity of convexity: Unsatisfiability is signaled when any one procedures signals unsatisfiable.

NO: Equational Theory Version

1. Equational theories are always consistent.
2. If $E \cup \{\exists x, y. x \neq y\}$ is consistent, then this theory is also stably-infinite.
3. Equational theories are convex. (If $E \vdash \phi \Rightarrow (l_1 \vee l_2)$, then consider the initial algebra induced by $E \cup \phi$ over an extended signature.)
4. Often decision procedures based on standard Knuth-Bendix completion can be used to deduce equalities.
5. Therefore, satisfiability procedures can be combined with only a polynomial time overhead.

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Application: Theory of Equality

$$\begin{array}{lcl} \Sigma & = & \Sigma_F \text{ (uninterpreted)} \\ \mathcal{T} & = & \text{Deductive closure of axioms of equality} \end{array}$$

- $\mathcal{T}$ is a stably-infinite equational theory.
- Congruence closure decides satisfiability of QFF in $\mathcal{T}$.
- $\therefore$ congruence closure for disjoint Σ_i can be combined in polynomial time.
- If congruence closure algorithm over a singleton Σ_i is described using completion, we get an abstract congruence closure for the combination.

Commutative Semigroup

$$\begin{array}{rcl} \Sigma & = & \{f\} \\ \mathcal{T} & = & \text{Axioms of equality + AC axioms for } f. \end{array}$$

- Treat f as variable arity

$$\begin{array}{rcll} f(\ldots, f(\ldots), \ldots) & = & f(\ldots, \ldots, \ldots) & \qquad (F) \\ f(\ldots, x, y, \ldots) & = & f(\ldots, y, x, \ldots) & \qquad (P) \end{array}$$

- Flatten all equations and do completion modulo P

$$\frac{f(c_1, c_1) \to c_1 \qquad f(c_1, c_2) \to f(c_2, c_2)}{f(c_1, c_2) = f(c_1, c_2, c_2)}$$

Commutative Semigroup

- All rules are of the form $f(\ldots) \rightarrow f(\ldots)$.
- Collapse guarantees termination of completion via Dickson's lemma.

$$\frac{f(c_1, c_1, c_2) \rightarrow c_1 \qquad f(c_1, c_2) \rightarrow c_1}{f(c_1, c_1, c_2) = c_1}$$

- Using an appropriate ordering on multisets, we get a algorithm to construct convergent systems (and decide satisfiability of QFF).

Example: Commutative Semigroup

If $E_0 = \{c_1^2 c_2 = c_3,\ c_1 c_2^2 = c_1 c_2\}$, we can use orientation, superposition (modulo AC), collapse to get a convergent (modulo AC) rewrite system

$$c_1^2 c_2 \to c_3,\ c_1 c_2^2 \to c_1 c_2$$

$$c_2 c_3 = c_1^2 c_2$$

$$c_1^2 c_2 \to c_2 c_3$$

$$c_3 = c_2 c_3$$

$$c_2 c_3 \to c_3$$

$$c_1^2 c_3 \to c_3^2$$

Application: Ground AC-theories

$$
\begin{array}{rcl}
\Sigma & = & \Sigma_F \cup \Sigma_{AC} \\
\mathcal{T} & = & \text{Axioms of equality + AC axioms for each } f \in \Sigma_{AC}.
\end{array}
$$

- Use Extension inference rule to purify equations
- Use abstract congruence closure on $\Sigma - \Sigma_{AC}$
- Use completion modulo AC on each $\{f\}$, $f \in \Sigma_{AC}$
- Combine by sharing equations between constants

Time Complexity: $O(n^2 * (T_{AC}(n) + n\log(n)))$.

Similarly, ACU-symbols can be added.

Gröbner Bases

$$\begin{array}{rcl} \Sigma & = & \{0, 1, +, \cdot, X_1, \ldots, X_n\} \cup \mathbb{Q} \\ \mathcal{T} & = & \text{Polynomial ring } \mathbb{K}[X_1, \ldots, X_n] \text{ over field } \mathbb{K} \end{array}$$

- Given a finite set of polynomial equations, new equations (between variables) can be deduced using Gröbner basis construction.
- Main inference rules is superposition. For e.g.,

$$\frac{c_1^2 c_2 \to 0 \qquad c_1 c_2^2 \to 1}{c_2 \cdot 0 = c_1 \cdot 1}$$

The equations are simplified and oriented s.t. the maximal monomial occurs on LHS, for e.g., $c_1 \to 0$.

Gröbner Bases: Contd

- Collapse simplifies LHS of rewrite rules.

$$\frac{c_1 \rightarrow 0 \qquad c_1 c_2^2 \rightarrow 1}{0 \cdot c_2^2 = 1}$$

  which simplifies to $0 = 1$, a contradiction.
- Using suitable ordering on monomials and sums of monomials, a convergent rewrite system (modulo the polynomial ring axioms), called a Gröbner basis, can be constructed in finite steps.
- Termination is established using Dickson's lemma as before.

Application: Gröbner Bases Plus . . .

$$\begin{array}{rcl} \Sigma & = & \Sigma_F \cup \Sigma_{AC} \cup \Sigma_{ACU} \cup \Sigma_{GB} \\ \mathcal{T} & = & \text{Union of the respective theories} \end{array}$$

Use NO combination, with the following decision procedures to deduce equalities:

- Use abstract congruence closure on $\Sigma - \Sigma_{AC}$
- Use completion modulo AC on each $\{f\}$, $f \in \Sigma_{AC}$
- Use completion modulo ACU on each $\{f\}$, $f \in \Sigma_{ACU}$
- Use Gröbner basis algorithm on equations over Σ_{GB}

Since each theory is convex and stably-infinite, we get a polynomial time combination over the individual theories.

Summary

The Nelson-Oppen theorem combines satisfiability procedures for conjunctions of literals in disjoint and stably-infinite theories.

- This is equivalent to deciding the validity of clauses: $\mathcal{T} \vdash \forall\vec{x}.(\phi_1 \Rightarrow \phi_2)$ where ϕ_1/ϕ_2 are AND/OR of atomic formulas.
- Using Purification, it is easy to see that we can restrict ϕ_2 to contain atomic formulae over variables.
- By definition, if $\mathcal{T}$ is convex and $=$ is the only predicate symbol, then validity above is equivalent to horn validity: $\mathcal{T} \vdash \forall\vec{x}.(\phi_1 \Rightarrow x_1 = x_2)$. This motivates the definition of convexity.

Summary

- Convexity allows optimization.
  - Convexity is also necessary for completeness of deterministic version of the NO procedure.
  - In the second part, additional assumptions grouped under the name Shostak theories, will allow for further optimized implementations of the deterministic NO procedure.
- Stably-infiniteness is required for completeness, i.e., if the component procedures return satisfiable, it allows construction of the fusion model.

Special Case: Theory with UIFs

Theorem 1 *Let $\mathcal{T}_1$ be a theory over a signature Σ. Let Σ_F be a disjoint set of function symbols with pure theory $\mathcal{T}_2$ of equality over it. If satisfiability of (quantifier-free) conjunction of literals can be decided in $O(T_1(n))$ time in $\mathcal{T}_1$, then,*

1. *The combined theory $\mathcal{T}$ is consistent.*
2. *Satisfiability of (quantifier-free) conjunction of literals in $\mathcal{T}$ can be decided in $O(2^{n^2} * (T_1(n) + n\log(n)))$ time.*
3. *If $\mathcal{T}_1$ and $\mathcal{T}_2$ are convex, then so is $\mathcal{T}$ and satisfiability in $\mathcal{T}$ is in $O(n^4 * (T_1(n) + n\log(n)))$ time.*

Single Theory with UIFs

- We modify the deterministic and non-deterministic procedures as follows:
  - purification is applied until all disequations over terms in Σ_2 are reduced to disequations between variables
  - all variables introduced by purification are considered shared between the two theories
  - rest is identical to the NO procedure
- Stably-infiniteness was required to get a bijection between the two models. Since there exist models of any cardinality, above a minimum which is communicated to $\mathcal{T}_1$, in $\mathcal{T}_2$, completeness holds.

Combination for the Word Problem

The word problem concerns with validity of an atomic formula.

- NO result can be modified to give a modularity result for this case.
- NO result can not be used as such, because the generated satisfiability checks may not be equivalent to word problems.
- If E_1 and E_2 are non-trivial equational theories over disjoint signatures with decidable word problems, then the word problem for $E_1 \cup E_2$ is decidable with a polynomial time overhead.

Non-Disjoint Signatures

Word problem in the union may not be decidable

E	:	semigroup presentation with undecidable word problem
E_1	:	Theory induced by E, with $\cdot$ uninterpreted (decided by congruence closure).
E_2	:	Theory of semigroups (decided by flattening).

Satisfiability in the union may not be decidable

E_1	:	$\{f(x, f(y, z)) = g(x, y, z)\}$
E_2	:	$\{f(f(x, y), z) = g(x, y, z)\}$
E	:	Theory of semi-groups

Non-Disjoint Signatures

- If $\mathbb{A}$ is a model for theory $\mathcal{T}_1 \cup \mathcal{T}_2$, then $\mathbb{A}^{\Sigma_1}$ and $\mathbb{A}^{\Sigma_2}$ is a model for $\mathcal{T}_1$ and $\mathcal{T}_2$ respectively.
- Define fusion of models $\mathbb{A}_1$ and $\mathbb{A}_2$ s.t. converse hold as well.
- Define a bijection between A_1 and A_2 and give interpretations accordingly.
- Generalize "stably-infiniteness": Identify conditions under which two models can be fused.
- Kinds of assumptions:
  - $\mathcal{T}_1^{\Sigma_1 \cap \Sigma_2}$ is identical to $\mathcal{T}_2^{\Sigma_1 \cap \Sigma_2}$
  - $\Sigma_1 \cap \Sigma_2$, or a subset thereof, generates both A_2 and A_2
  - Examples. Theories which admit constructors

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Shostak theories

- A ***canonizable*** and ***solvable*** theory is a ***Shostak theory***
- A ***canonizer*** σ maps terms to normal form terms s.t. equal terms in the theory are mapped to same form.
- A ***s***olver $solve$ maps an equation to an equivalent substitution.
- e.g., linear arithmetic
  - Canonizer returns ordered sum-of-monomials
  - Rational solver isolates, say, largest variable through scaling and cancellation.
  - Integral solver based on Euclid's algorithm

$$euclid(3x + 5y = 1) = \{x = -3 + 5k,\ y = 2 - 3k\}$$

where k is *fresh*.

Canonizable Theories

- A theory $\mathcal{T}$ is said to be ***canonizable*** if there is a computable $\sigma(a)$ such that
  - $\models_{\mathcal{T}} a = b$ iff $\sigma(a) \equiv \sigma(b)$
  - $vars(\sigma(a)) \subseteq vars(a)$
  - $\sigma(b) \equiv b$ for every subterm b of $\sigma(a)$
- A term a is said to be ***canonical*** if

$$\sigma(b) \equiv b$$

- Canonizer for linear arithmetic

$$\sigma_{\mathcal{A}}(y + x + x) \equiv 2x + y$$

Equality Sets

- An ***equality set*** E is of the form $\{a_1 = b_1, \dots, a_n = b_n\}$
- E is ***functional*** if $a = b_1, a = b_2 \in E$ implies $b_1 \equiv b_2$

Lookup: $E(a) := \begin{cases} b & : \quad a = b \in E \\ a & : \quad \text{otherwise} \end{cases}$

Apply:
$$\begin{aligned} E[x] &:= E(x) \\ E[f(a_1, \dots, a_n)] &:= E(f(E[a_1], \dots, E[a_n])) \end{aligned}$$

- A ***solution set*** is a functional equality set of the form

$$\{x_1 = b_1, \dots, x_n = b_n\}$$

with $x_i \notin vars(b_j)$ for $1 \leq i, j \leq n$

Preservation

- A variable assignment ρ' ***e**xtends* ρ if
  - $dom(\rho) \subseteq dom(\rho')$ and
  - $\rho(x) = \rho'(x)$ for all $x \in dom(\rho)$
- Let ψ, ψ' be sets of literals; then: ψ' ***$\mathcal{T}$-preserves*** ψ if
  - $vars(\psi) \subseteq vars(\psi')$
  - for all $\mathcal{T}$-interpretations $\mathbb{M}$ and assignments ρ there is some ρ' extending ρ such that

$$\mathbb{M}, \rho \models_{\mathcal{T}} \psi \;\text{ iff }\; \mathbb{M}, \rho' \models_{\mathcal{T}} \psi'$$

- In this case: $\models_{\mathcal{T}} \psi \Rightarrow \phi$ iff $\models_{\mathcal{T}} \psi' \Rightarrow \phi$
- This notion of preservation is sufficient for our purposes, since no new function symbols introduced.

Solvable Theories

- A theory $\mathcal{T}$ is called ***solvable*** if there is a computable procedure $solve(a = b)$
  - $solve(a = b) = \bot$ iff $a = b$ is $\mathcal{T}$***-unsatisfiable***
  - Otherwise, $solve(a = b) = S$, where S is a (functional) solution set such that
    - $dom(S) \subseteq vars(a = b)$
    - S $\mathcal{T}$***-preserves*** $a = b$
- Notice that fresh variables, that is, variables not in $vars(a = b)$ might be introduced on right-hand sides.

Theory of Lists

- **Signature.**

$$\Sigma_{\mathcal{L}} := \{cons(.,.), car(.), cdr(.)\}$$

- **Theory** $\mathcal{L}$ of lists contains the initial models of:

$$\begin{aligned} car(cons(a,b)) &= a \\ cdr(cons(a,b)) &= b \\ cons(car(a), cdr(a)) &= a \end{aligned}$$

- **Canonizer.**

  $\sigma_{\mathcal{L}}(a)$ is the normal form of the terminating and confluent TRS above.

List Solver

A configuration (E, S) consists of an equality set E and a solution set S

Decom $$\frac{(cons(a,b)=c \cup E,\ S)}{(\{a=car(c),\ b=cdr(c)\}\cup E,\ S)}$$

Solve $$\frac{(car(a)=b\cup E,S)}{(a=cons(b,k)\cup E,S)} \qquad \frac{(cdr(a)=b\cup E,S)}{(a=cons(k,b)\cup E,S)}$$ k fresh

VarElim $$\frac{(x=a \cup E,\ S)}{(\sigma_{\mathcal{L}}(E[x:=a]),\ S\circ\{x=a\})}$$ $x \notin vars(a)$

Triv $$\frac{(a=a \cup E,\ S)}{(E,\ S)}$$

Bot $$\frac{(a=b \cup E,\ S)}{\perp}$$ $a \not\equiv b$, $b \in [\![a]\!]$

where $S \circ S' := S' \cup \{x = \sigma_{\mathcal{L}}(S'[b]) \mid x = b \in S\}$

Booleans

- **Signature.**

  $$\Sigma_{\mathcal{B}} := \{true, false, ITE(., ., .)\}$$

- **Canonizer** $\sigma_{\mathcal{B}}$ returns, e.g., a binary decision diagrams (ordering on variables needed)

- **Solver.**

  process $a \iff b$ instead of $a = b$

$$\begin{array}{rcl} solve(true) & = & \{\} \\ solve(false) & = & \bot \\ solve(ITE(x, p, n)) & = & \{x = (p \wedge (n \Rightarrow \delta))\} \\ & & \cup\; solve(p \vee n) \end{array}$$

  where the δ's are fresh

Example: Boolean Solver

$$
\begin{aligned}
& solve(x \wedge y = \neg x) \\
= \quad & solve(ITE(x, ITE(y, false, true), false)) \\
= \quad & \{x = true\} \cup solve(ITE(y, false, true)) \\
= \quad & \{x = true,\ y = false\}
\end{aligned}
$$

Deciding a Shostak Theory

- Let $\mathcal{T}$ be a ***Shostak theory*** with canonizer $\sigma_{\mathcal{T}}(.)$ and solver $solve_{\mathcal{T}}(.)$
- We consider the validity problem

$$\models_{\mathcal{T}} E \Rightarrow a = b$$

- ***Template*** for decision procedure
  1. Build a solution set $S := process(id_E, E)$ using a finite number of $\mathcal{T}$-preserving transformations.
  2. Compute canonical forms $a' := S\langle\!\langle a \rangle\!\rangle$, $b' := S\langle\!\langle b \rangle\!\rangle$
  3. If $a' \equiv b'$ then ***Yes*** else ***No***

Deciding a Shostak Theory (Cont.)

- ***Canonization.***

$$S\langle\!\langle a\rangle\!\rangle := \sigma_{\mathcal{T}}(S[a])$$

- ***Fusion.***

$$S \rhd R := \{a = R\langle\!\langle b\rangle\!\rangle \mid a = b \in S\}$$

- ***Composition.***

$$\begin{array}{lcl} S \circ \bot & := & \bot \\ \bot \circ S & := & \bot \\ S \circ R & := & R \cup (S \rhd R) \end{array}$$

- For solved forms, $S \circ S = S$

Deciding a Shostak Theory (Cont.)

- Configuration (S, E) consists of an equality set E and a solution set S
- Building a solution set

$$\begin{array}{rcl} process(S, \emptyset) & = & S \\ process(\bot, E) & = & \bot \\ process(S,\ a = b \ \cup \ E) & = & process(assert(a = b,\ S),\ E) \\ assert(a = b,\ S) & = & S \circ solve(S\langle\!\langle a \rangle\!\rangle = S\langle\!\langle b \rangle\!\rangle) \end{array}$$

- For $S' = process^*(E, id_E)$

$$\models_{\mathcal{T}} (E \Rightarrow a = b)$$

iff

either $S' = \bot$ or $S'\langle\!\langle a \rangle\!\rangle \equiv S'\langle\!\langle b \rangle\!\rangle$

Soundness and Completeness

Let $S' := process(id_E, E)$;

- S' $\mathcal{T}$-preserves E, that is for every $\mathcal{T}$-interpretation $\mathbb{M}$ and an assignment ρ

  $\mathbb{M}, \rho \models E$ iff there is a ρ' extending ρ (to the variables in $vars(S')$) such that $\mathbb{M}, \rho' \models S'$

- ***Soundness.*** If $\sigma_{\mathcal{T}}(S'[a]) \equiv \sigma_{\mathcal{T}}(S'[b])$, then

$$\mathbb{M}, \rho' \models S' \Rightarrow a = S'[a] = \sigma_{\mathcal{T}}(S'[a]) = \sigma_{\mathcal{T}}(S'[b]) = S'[b] = b$$

  Thus, $\mathbb{M}, \rho \models E \Rightarrow a = b$.

- ***Completeness.*** By contraposition. Construct a model $\mathbb{M}, \theta$ such that $\mathbb{M}, \theta \models E$ but $\mathbb{M}, \theta \not\models a = b$.

Soundness and Completeness (Cont.)

When $\sigma_{\mathcal{T}}(S'[a]) \not\equiv \sigma_{\mathcal{T}}(S'[b])$

- there is a $\mathcal{T}$-model $\mathbb{M}, \theta$ s.t $\mathbb{M}, \theta \not\models S'[a] = S'[b]$
- $x \neq S'(x)$ for variables x in $S'[.]$
- Extend θ to an assignment θ' s.t

  $\theta'(x) := \mathbb{M}[\![S'(x)]\!]\theta$ if $x \neq S'(x)$

- $$\begin{aligned} \mathbb{M}, \theta' &\models S' \\ \mathbb{M}, \theta' &\models a = S'[a],\ S'[b] = b \end{aligned}$$
- Since S' $\mathcal{T}$-preserves (id_E, E), $\mathbb{M}, \theta \models E$ but $\mathbb{M}, \theta \not\models a = b$.

Shostak Theories in a NO Loop

- The solver and canonizer can be used to decide satisfiability of $E \cup D$ of equalities E and disequalities D in convex Shostak theories.
- *One way* to do this:
  - let $S' = process(id_{vars(E) \cup vars(D)}, E)$;
  - if $S' = \perp$ then return ***unsatisfiable***
  - if there is a disequality $a{\neq}b$ in D s.t. $S'\langle\langle a\rangle\rangle \equiv S'\langle\langle b\rangle\rangle$ then return ***unsatisfiable***.
  - Return ***satisfiable*** (and set of newly inferred variable equalities).
- Thus, convex and stably-infinite Shostak theories can be integrated with other disjoint, convex, stably-infinite theories using the NO result.

Outline

- Abstract Congruence Closure
- Nelson-Oppen Combination (NO)
  - Various Applications of NO
  - Shostak Theories
- Shostak Combination
- Commented Bibliography

Combining Shostak Theories

- **Problem.** Combination of the theory $\mathcal{T}_0$ of equality over UIF with several disjoint Shostak theories $\mathcal{T}_1,\ldots,\mathcal{T}_n$.
- Let σ_i and $solve_i$ be the canonizer and solver for theory $\mathcal{T}_i$.
- A term $f(a_1, \ldots, a_n)$ is an i-term if $f \in \Sigma_i$.
- A term a is a pure i-term if every subterm b of a is an i-term.

Composable Shostak Theories

- Resolve possible semantic incompatibilities between Shostak theories.
- ***Canonical Term model***
  - $D_{\mathcal{T}} := \{a \mid \sigma_{\mathcal{T}}(a) \equiv a\}$
  - $\mathbb{M}_{\mathcal{T}}(f)(a_1, \ldots, a_n) := \sigma_{\mathcal{T}}(f(a_1, \ldots, a_n))$
- ***Example.*** $\mathcal{T}_{CDS} := \{a \neq b, \forall x.x = a \vee x = b\}$ is canonizable ($\sigma(a) = a,\ \sigma(b) = b,\ \sigma(x) = x$) and solvable. Its canonical term model is $\{a, b, x_1, \ldots\}$ but it is only satisfiable in a two-element model.
- A Shostak theory $\mathcal{T}$ is ***composable*** if the canonical model $\mathbb{M}_{\mathcal{T}}$ is (isomorphic to) a $\mathcal{T}$-model.
- $\mathcal{T}$-validity is convex for composable Shostak theories

Convexity of Composable Theories

For a composable Shostak theory

$$\models_{\mathcal{T}} E \Rightarrow c_1 = d_1 \vee \ldots \vee c_n = d_n \text{ implies}$$
$$\models_{\mathcal{T}} E \Rightarrow c_k = d_k \text{ for some } k.$$

- Let $S := process(id_E, E)$ ($\mathcal{T}$-preserving)
- (wlog $S \neq \bot$) assume $\not\models_{\mathcal{T}} E \Rightarrow c_k = d_k$ for all k
- then $\not\models_{\mathcal{T}} S \Rightarrow c_k = d_k$
- Construct $\rho_S(x) := \begin{cases} \mathbb{M}_{\mathcal{T}}[\![S(x)]\!]\rho_S & S(x) \neq x \\ x & S(x) = x \end{cases}$
- $\mathbb{M}_{\mathcal{T}}, \rho_S \models S$
- $\mathbb{M}_{\mathcal{T}}, \rho_S \not\models c_k = d_k$ for all k (because $\mathbb{M}_{\mathcal{T}}[\![c_k]\!]\rho_S = \sigma(S[c_k])$)
- $\mathbb{M}_{\mathcal{T}}, \rho_S \not\models S \Rightarrow c_k = d_k$ for all k

Is this good or bad news?

- Shostak's algorithm is complete for Shostak theories $\mathcal{T}$ but a Shostak-like algorithm is not complete for the combination of $\mathcal{T} \cup \mathcal{T}_{UIF}$.
- Consider a Shostak theory with nonconvex $\mathcal{T}$-validity.
- Then $\models_{\mathcal{T}} E \Rightarrow a_1 = b_1 \vee \ldots \vee a_n = b_n$
- but $\not\models_{\mathcal{T}} E \Rightarrow a_i = b_i$ for $1 \leq i \leq n$.
- Consider:

$$\models_{\mathcal{T}} E, \begin{array}{ccc} f(a_1) = c & ,\ldots, & f(a_n) = c, \\ f(b_1) = d & ,\ldots, & f(b_n) = d \end{array} \Rightarrow c = d$$

- Can be shown to hold by case-splitting, which Shostak does not do…

Combining Canonizers

- ...is easy: Treat alien terms as variables and apply σ_i to canonize $f(a)$ when $f \in \mathcal{T}_i$.
- Let π_i be a chosen bijective equality set between the set of variables X and $\{a | (\exists j : j \neq i \wedge a \in \mathcal{T}_j)\}$.
- Individual canonizers for impure terms

$$\sigma_i'(a) \;:=\; \pi_i[\sigma_i(a')], \text{ when } a' : \pi_i[a'] \equiv a$$

The combined canonizer

$$\begin{aligned} \sigma(x) &= x \\ \sigma(f_i(a_1, \ldots, a_n)) &= \sigma_i'(f_i(\sigma(a_1), \ldots, \sigma(a_n))) \end{aligned}$$

Combining Solvers: The Problem

…already shows up when combining Shostak theories

- Consider

$$5 + car(x + 2) = cdr(x + 1) + 3$$

  in $\mathcal{T}_{\mathcal{A}} \cup \mathcal{T}_{\mathcal{L}}$.
- The individual theories $\mathcal{T}_{\mathcal{A}}$ (arithmetic) and $\mathcal{T}_{\mathcal{L}}$ (lists) have solvers and canonizers.

The Problem (Cont.)

- Assume a combined solver which treats alien terms as variables and applies component solvers $solve_{\mathcal{A}}$ or $solve_{\mathcal{L}}$ according to the top-level symbol.
- ***Example***

$$
\begin{array}{rcl}
 & & 5+car(x+2) = cdr(x+1)+3 \\
(solve_{\mathcal{A}}) & \leadsto & car(x+2) = cdr(x+1)-2 \\
(solve_{\mathcal{L}}) & \leadsto & x+2 = cons(cdr(x+1)-2, k) \\
(solve_{\mathcal{A}}) & \leadsto & x = cons(cdr(x+1)-2, k)-2
\end{array}
$$

- but this is not a solved form: x occurs on the right.

The Solution

- Shostak theories can be combined without combining solvers
- Key ideas
  - Maintain theory-wise solution sets
  - Communicate variable equalities as in NO
  - Construct combined canonizer (as required in a Shostak combination)
- For $\mathcal{T}_{\mathcal{A}} \cup \mathcal{T}_{\mathcal{L}}$ configurations $S := (S_V, S_A, S_L)$ consist of
  - variable equalities S_V in canonical form
  - a solution set S_A for the theory $\mathcal{T}_{\mathcal{A}}$
  - a solution set S_L for the theory $\mathcal{T}_{\mathcal{L}}$

Process

$$\begin{array}{rcl} process(S;\emptyset) & := & S \\ process(S;\{a=b\}\cup T) & := & process(assert(S;a=b);T) \\ assert(S;a=b) & := & close^*(merge(abstract^*(S;S[\![a=b]\!]))) \end{array}$$

1. Canonize $a = b$ w.r.t. S to get $a' = b'$.
2. Variable abstract $a' = b'$: Replace $f(x_1, \ldots, x_n)$ by a fresh x, and adding $x = x$ to S_V and $x = f(x_1, \ldots, x_n)$ to S_i. Iteration yields $x = y$ from $a' = b'$.
3. Merge $x = y$ into S to yield $S_V \circ \{x = y\}$, assuming $x \prec y$.
4. Close S: When x, y such that
   - $S_i(x) \equiv S_i(y)$ but $S_V(x) \not\equiv S_V(y)$, merge $x = y$ into S.
   - $S_V(x) \equiv S_V(y)$ but $S_i(x) \not\equiv S_i(y)$, merge $solve(S_i(x) = S_i(y))$ into S_i

Example

- Variable abstract $5 + car(x+2) = cdr(x+1) + 3$ to $v_3 = v_6$

$$\left(\begin{array}{l} \{x = x, v_1 = v_1, v_2 = v_2, v_3 = v_6, v_4 = v_4, v_5 = v_5, v_6 = v_6\} \\ \{v_1 = x + 2, v_3 = v_2 + 5, v_4 = x + 1, v_6 = v_5 + 3\} \\ \{v_2 = car(v_1), v_5 = cdr(v_4)\} \end{array}\right)$$

- Since v_3 and v_6 are merged in S_V but not in S_A, solve $S_A(v_3) = S_A(v_6)$ in A.

$$solve_A(v_2 + 5 = v_5 + 3) \ = \ \{v_2 = v_5 - 2\}$$

Example (Cont.)

…Result of solve was $\{v_2 = v_5 - 2\}$

- Compose result

$$\left(\begin{array}{l} \{x = x, v_1 = v_1, v_2 = v_2, v_3 = v_6, v_4 = v_4, v_5 = v_5, v_6 = v_6\} \\ \{v_1 = x + 2, v_3 = v_5 + 3, v_4 = x + 1, v_6 = v_5 + 3, v_2 = v_5 - 2\} \\ \{v_2 = car(v_1), v_5 = cdr(v_4)\} \end{array}\right)$$

- No new variable equalities to be propagated.
- The different solved forms of both v_2 and v_5 are tolerated, since canonizer picks a solution that is appropriate to the context.

Example (Cont.)

- Canonical state

$$\left(\begin{array}{l}\{x = x, v_1 = v_1, v_2 = v_2, v_3 = v_6, v_4 = v_4, v_5 = v_5, v_6 = v_6\} \\ \{v_1 = x + 2, v_3 = v_5 + 3, v_4 = x + 1, v_6 = v_5 + 3, v_2 = v_5 - 2\} \\ \{v_2 = car(v_1), v_5 = cdr(v_4)\}\end{array}\right)$$

- $5 + car(x + 2) \leadsto 5 + car(v_1) \leadsto 5 + v_2 \leadsto 3 + v_5 \leadsto v_6$
- $cdr(x + 1) + 3 \leadsto cdr(v_4) + 3 \leadsto v_5 + 3 \leadsto v_6$

Canonizer

- σ_i is only defined for pure i-terms.
- σ'_i is the extension of σ_i that deals with alien terms by treating them as variables.
- Canonizer for the combination of Shostak theories $\mathcal{T}_i$.

$$
\begin{array}{rcl}
S[\![x]\!] & = & S_V(x) \\
S[\![f_i(a_1,\ldots,a_n)]\!] & = & S_V(x), \text{ when} \\
 & & x = \sigma'_i(f_i(S_i(S[\![a_1]\!]),\ldots,S_i(S[\![a_n]\!]))) \in S_i \\
S[\![f_i(a_1,\ldots,a_n)]\!] & = & \sigma'_i(f_i(S_i(S[\![a_1]\!]),\ldots,S_i(S[\![a_n]\!])))
\end{array}
$$

Congruence Closure Revisited

$$\begin{aligned} \Sigma &= \Sigma_F \text{ (uninterpreted)} \\ \mathcal{T} &= \text{Deductive closure of axioms of equality} \end{aligned}$$

- Validity problem $\models E \Rightarrow a = b$
- State consists of $(S_V;\ S_U;\ E)$
  - S_V contains the variable equalities $x = y$
  - S_U contains equalities $x = f(x_1, \ldots, x_n)$
  - E contains the unprocessed input equalities.
- $(S_V;\ S_U)$ together form the solution state S
- S_V partitions the variables into equivalence classes
- x, y are in the same equivalence class if $S_V(x)$ and $S_V(y)$

Template for Shostak CC

- Start state $S_0 := (id_E;\ \emptyset;\ E)$
- Compute $S^* = process(S_0)$ by iterating

$$\begin{array}{rcl} process(S;\emptyset) & = & S \\ process(S;\{a=b\} \cup E) & = & process(assert(S;a=b);E) \\ assert(S;a=b) & = & close^*(merge(abstract^*(S;S[\![a=b]\!]))) \end{array}$$

- Check canonical forms: $S^*[\![a]\!] \equiv S^*[\![b]\!]$
- Present treatment a specific strategy of ***abstract*** CC.

Congruence Closure Revisited (Cont.)

For each input equality $a = b$ and state S:

1. Canonize $a = b$ w.r.t. S to get $a' = b'$.
2. Variable abstract $a' = b'$: Replace $f(x_1, \ldots, x_n)$ by a fresh x, and adding $x = x$ to S_V and $x = f(x_1, \ldots, x_n)$ to S_U. Iteration yields $x = y$ from $a' = b'$.
3. Merge $x = y$ into S to yield $S_V \circ \{x = y\}$; $S_U \triangleright \{x = y\}$, assuming $x \prec y$.
4. Close S: When x, y, such that $S_U(x) \equiv S_U(y)$ but $S_V(x) \not\equiv S_V(y)$, merge $x = y$ into S.

Example

- Validity problem

$$\models \{f(f(f(x))) = x,\ x = f(f(x))\} \Rightarrow f(x) = x$$

- Start state

$$S_0 := (\{x = x\}; \emptyset; \{f(f(f(x))) = x,\ x = f(f(x))\}$$

- Abstraction

$$abstract(\{x = x\};\ \emptyset;\ f(f(f(x))) = x)$$

$$\rightsquigarrow \begin{pmatrix} \{x = x,\ v_1 = v_1,\ v_2 = v_2,\ v_3 = v_3\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(v_2)\} \\ v_3 = x \end{pmatrix}$$

Example (Cont.)

$$\begin{pmatrix} \{x = x,\ v_1 = v_1, v_2 = v_2,\ v_3 = v_3\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(v_2)\} \\ v_3 = x \end{pmatrix}$$

$$merge \ \rightsquigarrow \ \begin{pmatrix} \{x = x,\ v_1 = v_1,\ v_2 = v_2,\ {\color{red} v_3 = x}\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(v_2)\} \end{pmatrix}$$

Example (Cont.)

- Variables x, y are incongruent if
  - $S_V(x) \not\equiv S_V(y)$ and
  - $S_U(x) \equiv S_U(y)$
- There are no incongruences in our running example.

$$\left(\begin{array}{l} \{x = x,\ v_1 = v_1,\ v_2 = v_2,\ v_3 = x\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(v_2)\} \end{array} \right)$$

Example (Cont.)

$$\left(\begin{array}{l} \{x = x,\ v_1 = v_1,\ v_2 = v_2,\ v_3 = x\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(v_2)\} \end{array} \right)$$

Processing of $x = f(f(x))$. Canonization and orientation yield $v_2 = x$, which is merged

$$\left(\begin{array}{l} \{x = x,\ v_1 = v_1,\ v_2 = x,\ v_3 = x\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(x)\} \end{array} \right)$$

The incongruence between v_1, v_3 is fixed by close

$$S^* := \left(\begin{array}{l} \{x = x,\ v_1 = x,\ v_2 = x,\ v_3 = x\} \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(x)\} \end{array} \right)$$

Example (Cont.)

- Canonical form $S[\![a]\!]$ of a term a with respect to S

$$\begin{array}{rcl} S[\![x]\!] & = & S_V(x) \\ S[\![f(a_1,\ldots,a_n)]\!] & = & S_V(x),\ \text{when } x : x = f(S[\![a_1]\!],\ldots,S[\![a_n]\!]) \in S_U \\ S[\![f(a_1,\ldots,a_n)]\!] & = & f(S[\![a_1]\!],\ldots,S[\![a_n]\!]),\ \text{otherwise.} \end{array}$$

- Example

$$S^* := \left(\begin{array}{l} \{x = x,\ v_1 = x,\ v_2 = x,\ v_3 = x\}; \\ \{v_1 = f(x),\ v_2 = f(v_1),\ v_3 = f(x)\} \end{array} \right)$$

- Now, $S^*[\![f(x)]\!] \equiv x \equiv S^*[\![x]\!]$

Multi-Shostak

- Consider the union $\mathcal{T} = \bigcup_{i=0}^{n} \mathcal{T}_i$ of the equality theory of $\mathcal{T}_0$ for UIF and a set of disjoint, composable Shostak theories $\mathcal{T}_i$ $(i = 1, \ldots, n)$
- An I-model of $\mathcal{T}$ is a model $\mathbb{M}$ whose reduct w.r.t $\mathcal{T}_i$ is a $\mathcal{T}_i$-model for every $i = 1, \ldots, n$.
- Validity problem $\models_I E \Rightarrow a = b$

Multi-Shostak: Process

Decision procedure

1. Compute $S^* := process(id_E; E)$

$$\begin{array}{rcl} process(S;\ \emptyset) & = & S \\ process(S;\ E) & = & \bot,\ \text{when } i : S_i = \bot \\ process(S;\ \{a = b\} \cup E) & = & process(assert(S; a = b);\ E) \\ assert(S; a = b) & = & close^*(merge_V(abstract^*(S;\ a' = b'))) \\ & & \text{where } a' = S[\![a]\!], b' = S[\![b]\!] \end{array}$$

2. If $S[\![a]\!] \equiv S[\![b]\!]$ then ***Yes*** else ***No***

Canonical Solution States

- Invariants
  - S_V is functional and idempotent
  - S_0 is functional and normalized ($S_0 \rhd S_V = S_0$)
  - S_i ($i > 0$) are (functional) solution sets, idempotent, normalized ($S_i \rhd S_V = S_i$)
- A solution state S is ***confluent*** if for all $x, y \in dom(S_V)$ and $0 \leq i \leq N$:

$$S_V(x) \equiv S_V(y) \iff S_i(x) \equiv S_i(y)$$

- A ***canonical*** solution state S is confluent and satisfies the invariants above.

Multi-Shostak: Process

- *abstract*

  Replace maximal pure i-term c with fresh variable x, adding $x = c$ to S_i.

- $merge_V$

  $S_V;\ S_U;\ x = y \ \rightsquigarrow\ S_V \circ \{x = y\};\ S_U \triangleright \{x = y\}$

- $merge_i$

  $S_i;\ x = y \ \rightsquigarrow\ S_i \circ_i solve(S_i(x) = S_i(y))$

- $close(S)$

  Apply $merge_i$ or $merge_V$ to restore canonicity.

Multi-Shostak: Abstraction

$$
\begin{aligned}
abstract(S;\ x = y) &= (S;\ x = y), \\
abstract(S;\ a = b) &= (S';\ \{c = x\}[a] = \{c = x\}[b]) \text{ when } S', c, x, i: \\
&\quad (c \equiv f_0(x_1, \ldots, x_n) \text{ or } c \in max(\llbracket a = b \rrbracket_i)(i > 0)), \\
&\quad x \notin vars(S \cup a = b), \\
&\quad S'_V = S_V \cup \{x = x\}, \\
&\quad S'_i = S_i \cup \{x = c\}, \\
&\quad S'_j = S_j, \text{ for } , i \neq j
\end{aligned}
$$

- $max(\llbracket a = b \rrbracket_i)$ is a maximal pure i-term
- If $g(x)$ in $f(g(x))$ is replaced with y and $f(y)$ by z then { y = g(x), z = f(y) } is not idempotent ($i > 0$).

Multi-Shostak: Close

$$
\begin{array}{rcl}
close(S) & = & S, \text{ when } i : S_i = \bot_i \\
close(S) & = & S', \quad \text{when } S', i, x, y : \\
 & & \quad x, y \in dom(S_V), \\
 & & \quad (i > 0, S_V(x) \equiv S_V(y), S_i(x) \not\equiv S_i(y), \text{ and} \\
 & & \quad\ S' = merge_i(S;\ x = y)) \\
 & & \quad \text{or} \\
 & & \quad (i \geq 0, S_V(x) \not\equiv S_V(y), S_i(x) \equiv S_i(y), \text{ and} \\
 & & \quad\ S' = merge_V(S;\ S_V(x) = S_V(y))) \\
close(S) & = & normalize(S), \text{ otherwise.} \\
normalize(S) & = & (S_V;\ S_0;\ S_1 \rhd S_V;\ \ldots;\ S_N \rhd S_V).
\end{array}
$$

Multi-Shostak: Merge

$$
\begin{array}{rcl}
merge_i(S; x = y) & = & S', \text{ where } \; i > 0, \\
& & \qquad S'_i = S_i \circ_i solve(S_i(x) = S_i(y)), \\
& & \qquad S'_j = S_j, \text{ for } i \neq j, \\
& & \qquad S'_V = S_V. \\
merge_V(S; x = x) & = & S \\
merge_V(S; x = y) & = & (S_V \circ R;\ S_0 \triangleright R;\ S_1;\ \ldots;\ S_N) \\
& & \quad \text{where } R = orient(x = y).
\end{array}
$$

Multi-Shostak: Canonizer

Given a ***canonical*** state $S_V;\ S_0; \ldots;\ S_n$, a combined canonizer can be defined as:

$$
\begin{array}{rcl}
S[\![x]\!] & := & S_V(x) \\
S[\![f_i(a_1, \ldots, a_n)]\!] & := & S_V(x),\ \text{when} \\
 & & x = \sigma_i'(f_i(S_i(S[\![a_1]\!]), \ldots, S_i(S[\![a_n]\!]))) \in S_i \\
S[\![f_i(a_1, \ldots, a_n)]\!] & := & \sigma_i'(f_i(S_i(S[\![a_1]\!]), \ldots, S_i(S[\![a_n]\!])))
\end{array}
$$

with $\sigma_0(a) = a$ and $S_0(a) = a$.

Termination

- $S[\![a = b]\!]$ is terminating
- $abstract^*(S; a = b)$ is terminating
- $close^*(S)$ terminates, because the sum of the number of equivalence classes over variables in $dom(S_V)$ decreases in each iteration.

Soundness and Completeness

Theorem. Let $\mathcal{T}$ with signature Σ be the union of

- the theory $\mathcal{T}_0$ of UIF
- and $\mathcal{T}_i$ $(i = 1, \ldots, n)$ be disjoint, composable Shostak theories.

Furthermore, let

- $S^* := \mathit{process}^*(\mathit{id}_E; E)$ and
- $I = \{1 \ldots, n\}$;

then:

$$\models_I E \Rightarrow a = b$$

iff

either $S^* = \bot$ or $S^*[\![a]\!] \equiv S^*[\![b]\!]$

Proof Outline

- If $S' := process(id_E; E)$, then S' I-preserves E.
- **Soundness.** if $S[\![a]\!] \equiv S[\![b]\!]$, then

$$\models_I S' \Rightarrow a = S'[\![a]\!] = S'[\![b]\!] = b$$

  Thus, $\models_I E \Rightarrow a = b$
- **Completeness.** by contraposition:

  if $S'[\![a]\!] \not\equiv S'[\![b]\!]$ then $\not\models_I S' \Rightarrow a = b$

  for canonical S'.
- Construct an I-model $\mathbb{M}_{S'}, \rho_{S'}$ s.t.

  $M_{S'}, \rho_{S'} \models S'$ but $M_{S'}, \rho_{S'} \not\models a = b$

Canonical Term Model

- Definition
  - $D_{S'} := \{e \in \mathcal{T}(\Sigma, vars(S')) \mid S'[\![e]\!] \equiv e\}$
  - $\mathbb{M}_{S'}(f)(e_1, \ldots, e_n) := S'[\![f(e_1, \ldots, e_n)]\!]$
  - $\rho_{S'}(x) := S_V(x)$
- Properties
  - $\mathbb{M}_{S'}[\![c]\!]\rho_{S'} = S'[\![c]\!]$
  - $\mathbb{M}_{S'}, \rho_{S'} \models S'$
  - $\mathbb{M}_{S'}$ is an I-model, since $\mathbb{M}_{S'}$ is isomorphic to $\mathbb{M}_i$ for each i $(1 \leq i \leq n)$ and i is composable.
- Corollary: I-validity is convex.

Canonical Term Model (Cont.)

The canonical term model $\mathbb{M}_S$ is isomorphic to the canonical i-model $\mathbb{M}_i$

- The isomorphism μ_i is defined between D_S (all S-canonical terms) and D_i (all i-canonical terms) so that

$$\begin{array}{rcl} \mu_i(x) & = & a' \quad \text{where } \pi_i(a') = S_i(x) \\ \mu_i(f_i(\bar{b})) & = & f_i(\mu_i(\bar{b})) \\ \mu_i(f_j(\bar{b})) & = & \pi_i^{-1}(f_j(\bar{b})) \quad j \neq i \end{array}$$

- Need to show that $\mu_i(M_S(f_i)(a)) = M_i(f_i)(\mu_i(a))$ for $a \in D_S$

Summary

- Decision procedure based on Shostak's ideas for the combination of equality over UIF and disjoint, composable Shostak theories.
- Key idea: separate solution sets for individual theories.
- Variable dependencies can be cyclic across theories.
- Shostak combination an instance of NO combination.
- Added advantage is a global canonizer.

ICS: Integrated Canonizer and Solver

- A variant of the Shostak combination described here is implemented in ICS.
- The theory supported by ICS currently includes:
  - Equality and disequality.
  - Rational and integer linear arithmetic.
  - Theory of tuples, S-expressions
  - Boolean constants.
  - Array theory
  - Theory of bitvectors
- Available free of charge for noncommercial applications under the ICS license agreement.

  `ics.csl.sri.com`

Bibliography

- Armando, A., Ranise, S., and Rusinowitch, M., "*A rewriting approach to satisfiability procedure*", IC'02.
  deriving decision procedures
- Baader, F. and Tinelli, C., "*Deciding the word problem in the union of equational theories*", IC'02.
  theories sharing constructors
- Bachmair, L., Tiwari, A., and Vigneron, L., "*Abstract congruence closure*", JAR'02.
  Abstract CC, specializations, complexity
- Barrett, C. W., Dill, D. L., and Stump, A., "*A generalization of Shostak's method for combining decision procedures*", FroCoS'02.
  Shostak in NO procedure, convexity and stably-infiniteness

Bibliography

- Bjorner, N. S., "*Integrating decision procedures for temporal verification*", PhD Thesis'98.
  general results plus proedures for individual theories
- Cyrluk, D., Lincoln, P., and Shankar, N., "*On Shostak's decision procedure for combination of theories*", CADE'96.
  Shostak's CC, Single theory with UIF
- Downey, P. J., Sethi, R., and Tarjan, R. E., "*Variations on the common subexpression problem*", JACM'80.
  CC + linear variant
- Ganzinger, H., "*Shostak Light*", CADE 2002.
  Th + UIFs, convexity also necessary, stably-infiniteness not required, sigma-models indistinguishable

Bibliography

- Halpern, J. Y., "*Presburger arithmetic with unary predicates is Π^1_1-complete*", JSC'91.
  undecidability by adding predicates
- Kapur, D., "*Shostak's congruence closure as completion*", RTA'97.
  CC algorithm
- Kapur, D., "*A rewrite rule based framework for combining decision procedures*", FroCoS'02.
  Shostak combination
- Lynch, C. and Morawska, B., "*Automatic decidability*", LICS'02.
  deriving decision procedures and complexity

Bibliography

- Nelson, G. and Oppen, D., "*Simplification by cooperating decision procedures*", ACM TOPLAS'79.
  Combination result, specific theories
- Nelson, G. and Oppen, D., "*Fast decision procedures based on congruence closure*", JACM'80.
  CC, theory of lists
- Oppen, D. C., "*Complexity, convexity, and combination of theories*", TCS'80.
  NO main theorem, complexity, special theories
- Pratt, V. R., "Two easy theories whose combination is hard", MIT TR'77.
  validity hard for a combination of non-convex PTIME theories

Bibliography

- Rueß, H. and Shankar, N.,“*Deconstructing Shostak*”, LICS’01.
  Shostak theory + UIF–the Shostak way
- Shankar, N. and Rueß, H., “*Combining Shostak theories*”, RTA’02.
  Multiple Shostak theory combination
- Shostak, R. E., “*An efficient decision procedure for arithmetic with function symbols*”, SRI TR’77.
  arithmetic + UIFs
- Shostak, R. E., “*Deciding combinations of theories*”, JACM’84.
  Shostak theory + UIF

Bibliography

- Stump, A., Dill, D., Barrett, C., and Levitt, J., "*A decision procedure for extensional theory of arrays*", LICS'01.
  theory of arrays
- Tinelli, C. and Ringeissen, C., "*Unions of non-disjoint theories and combinations of satisfiability procedures*", Elveiser Science'01.
  New advances for non-disjoint combinations
- Tiwari, A., "*Decision procedures in automated deduction*", PhD Thesis'00.
  Shostak theories in NO framework