Seminar Decision Procedures and Applications

Background Information: Part I

Viorica Sofronie-Stokkermans

University Koblenz-Landau

25 June 2019

Topics for the talks

- **Matthias Becker:** Decision Procedures for UTVPI Constraints
- **Delzar Habash:** Automata approach to Presburger arithmetic
- **Denis Oldenburg:** Quantifier elimination for linear arithmetic over the integers
- **Dominik Kohns:** Reasoning about uninterpreted function symbols

- **Nico Bartmann:** DPLL(T)

- **Stefan Strüder:** Decision procedures for classical datatypes based on the superposition calculus
- **Tim Taubitz:** Instantiation-based decision procedures for theories of arrays.
- **Jouliet Mesto:** Data Structure Specifications via Local Equality Axioms.

Structure

Reasoning in standard theories

Presburger arithmetic: Delzar Habash, Denis Oldenburg

Simpler fragments: UTVPI Matthias Becker

Theory of uninterpreted function symbols: Dominik Kohns

Conjunctive fragment $\mapsto$ clauses: Nico Bartmann

Classical data types: Stefan Strüder: Superposition

Structure

Reasoning in complex theories

Modular reasoning in combinations of theories

Disjoint signature: The Nelson-Oppen method

- **Applications: complex data types**

  Fragment of theory of arrays: Tim Taubitz

  Fragment of theory of pointers: Jouliet Mesto

Logical theories

Syntactic view

Axiomatized by a set $\mathcal{F}$ of (closed) first-order Σ-formulae.

the models of $\mathcal{F}$: $\quad \text{Mod}(\mathcal{F}) = \{\mathcal{A} \in \Sigma\text{-alg} \mid \mathcal{A} \models G, \text{ for all } G \text{ in } \mathcal{F}\}$

$\mathcal{F} \subseteq \text{Th}(\text{Mod}(\mathcal{F}))$ (typically strict)

$\mathcal{M} \subseteq \text{Mod}(\text{Th}(\mathcal{M}))$ (typically strict)

Semantic view

given a class $\mathcal{M}$ of Σ-structures

the first-order theory of $\mathcal{M}$: $\text{Th}(\mathcal{M}) = \{G \in F_\Sigma(X) \text{ closed} \mid \mathcal{M} \models G\}$

$\text{Th}(\text{Mod}(\mathcal{F}))$ the set of formulae true in all models of $\mathcal{F}$

represents exactly the set of consequences of $\mathcal{F}$

Examples

1. Linear integer arithmetic. $\Sigma = (\{0/0, s/1, +/2\}, \{\leq /2\})$

$\mathbb{Z}_+ = (\mathbb{Z}, 0, s, +, \leq)$ the standard interpretation of integers.

$\{\mathbb{Z}_+\} \subset \mathsf{Mod}(\mathsf{Th}(\mathbb{Z}_+))$

2. Uninterpreted function symbols. $\Sigma = (\Omega, \mathsf{Pred})$

$\mathcal{M} = \Sigma$-alg: the class of all Σ-structures

The theory of uninterpreted function symbols is Th(Σ-alg)
the family of all first-order formulae which are true in all Σ-structures.

Examples

3. Lists. $\Sigma = (\{\mathsf{car}/1, \mathsf{cdr}/1, \mathsf{cons}/2\}, \emptyset)$

$$\mathcal{F} = \left\{ \begin{array}{rcl} \mathsf{car}(\mathsf{cons}(x, y)) & \approx & x \\ \mathsf{cdr}(\mathsf{cons}(x, y)) & \approx & y \\ \mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x)) & \approx & x \end{array} \right.$$

$\mathsf{Mod}(\mathcal{F})$: the class of all models of $\mathcal{F}$

$\mathsf{Th}_{\mathsf{Lists}} = \mathsf{Th}(\mathsf{Mod}(\mathcal{F}))$ theory of lists (axiomatized by $\mathcal{F}$)

Decidable theories

$\Sigma = (\Omega, \mathsf{Pred})$ be a signature.

$\mathcal{M}$: class of Σ-structures. $\mathcal{T} = Th(\mathcal{M})$ is decidable

iff

there is an algorithm which, for every closed first-order formula ϕ, can decide (after a finite number of steps) whether ϕ is in $\mathcal{T}$ or not.

$\mathcal{F}$: class of (closed) first-order formulae.

The theory $\mathcal{T} = Th(Mod(\mathcal{F}))$ is decidable

iff

there is an algorithm which, for every closed first-order formula ϕ, can decide (in finite time) whether $\mathcal{F} \models \phi$ or not.

Examples

Undecidable theories

- Peano arithmetic

Axiomatized by:

$\forall x \neg(x + 1 \approx 0)$ (zero)

$\forall x \forall y\, (x + 1 \approx y + 1 \rightarrow x \approx y$ (successor)

$F[0] \wedge (\forall x\, (F[x] \rightarrow F[x + 1]) \rightarrow \forall x F[x])$ (induction)

$\forall x\, (x + 0 \approx x)$ (plus zero)

$\forall x, y\, (x + (y + 1) \approx (x + y) + 1)$ (plus successor)

$\forall x, y\, (x * 0 \approx 0)$ (times zero)

$\forall x, y\, (x * (y + 1) \approx x * y + x)$ (times successor)

$3 * y + 5 > 2 * y$ expressed as $\exists z(z \neq 0 \wedge 3 * y + 5 \approx 2 * y + z)$

Intended interpretation: $(\mathbb{N}, \{0, 1, +, *\}, \{\approx, \leq\})$

(does not capture true arithmetic by Gödel's incompleteness theorem)

- $\text{Th}((\mathbb{Z}, \{0, 1, +, *\}, \{\leq\}))$
- $\text{Th}(\Sigma\text{-alg})$

Examples

In order to obtain decidability results:

- Restrict the signature
- Enrich axioms
- Look at certain fragments

Examples

In order to obtain decidability results:

- Restrict the signature
- Enrich axioms
- Look at certain fragments

Decidable theories

- Presburger arithmetic decidable in 3EXPTIME [Presburger'29]
  Signature: $(\{0, 1, +\}, \{\approx, \leq\})$ (no $*$)

  Axioms { (zero), (successor), (induction), (plus zero), (plus successor) }

A decision procedure will be presented by Delzar Habash

A quantifier-elimination method with be presented by Denis Oldenburg

A simple fragment (UTVPI) with be presented by Matthias Becker

Examples

In order to obtain decidability results:

- Restrict the signature
- Enrich axioms
- Look at certain fragments

Decidable theories

- The theory of real numbers (with addition and multiplication) is decidable in 2EXPTIME [Tarski'30]

Examples

In order to obtain decidability results:

- Restrict the signature
- Enrich axioms
- Look at certain fragments $\mathcal{L} \subseteq \mathsf{Fma}(\Sigma)$

  "Simpler" task: Given ϕ in $\mathcal{L}$, is it the case that $\mathcal{T} \models \phi$?

Common restrictions on $\mathcal{L}$

	Pred $= \emptyset$	$\{\phi \in \mathcal{L} \mid \mathcal{T} \models \phi\}$
$\mathcal{L}=\{\forall x A(x) \mid A \text{ atomic}\}$	word problem	
$\mathcal{L}=\{\forall x (A_1 \wedge \ldots \wedge A_n \rightarrow B) \mid A_i, B \text{ atomic}\}$	uniform word problem	$\mathsf{Th}_{\forall \mathsf{Horn}}$
$\mathcal{L}=\{\forall x C(x) \mid C(x) \text{ clause}\}$	clausal validity problem	$\mathsf{Th}_{\forall,\mathsf{cl}}$
$\mathcal{L}=\{\forall x \phi(x) \mid \phi(x) \text{ unquantified}\}$	universal validity problem	$\mathsf{Th}_{\forall}$

Validity of ∀ formulae vs. ground satisfiability

The following are equivalent:

(1) $\mathcal{T} \models \forall x(L_1(x) \vee \cdots \vee L_n(x))$

(2) There is no model of $\mathcal{T}$ which satisfies $\exists x(\neg L_1(x) \wedge \cdots \wedge \neg L_n(x))$

(3) There is no model of $\mathcal{T}$ and no valuation for the constants c

for which $(\neg L_1(c) \wedge \cdots \wedge \neg L_n(c))$ becomes true

(notation: $(\neg L_1(c) \wedge \cdots \wedge \neg L_n(c)) \models_{\mathcal{T}} \perp$)

Can reduce any validity problem to a ground satisfiability problem

Useful theories

Many example of theories in which ground satisfiability is decidable:

- The empty theory (no axioms) $UIF(\Sigma)$: Dominik Kohns
- theories axiomatizing common datatypes: Stefan Strüder

Combination of theories

Combinations of theories and models

Forgetting symbols

Let $\Sigma = (\Omega, \Pi)$ and $\Sigma' = (\Omega', \Pi')$ s.t. $\Sigma \subseteq \Sigma'$, i.e., $\Omega \subseteq \Omega'$ and $\Pi \subseteq \Pi'$

For $\mathcal{A} \in \Sigma'$-alg, we denote by $\mathcal{A}_{|\Sigma}$ the Σ-structure for which:

$$U_{\mathcal{A}_{|\Sigma}} = U_{\mathcal{A}}, \quad f_{\mathcal{A}_{|\Sigma}} = f_{\mathcal{A}} \text{ for } f \in \Omega; \quad P_{\mathcal{A}_{|\Sigma}} = P_{\mathcal{A}} \text{ for } P \in \Pi$$

(ignore functions and predicates associated with symbols in $\Sigma' \backslash \Sigma$)

$\mathcal{A}_{|\Sigma}$ is called the restriction (or the reduct) of $\mathcal{A}$ to Σ.

Example: $\Sigma' = (\{+/2, */2, 1/0\}, \{\leq /2, \text{even}/1, \text{odd}/1\})$

$\Sigma = (\{+/2, 1/0\}, \{\leq /2\}) \subseteq \Sigma'$

$\mathcal{N} = (\mathbb{N}, +, *, 1, \leq, \text{even}, \text{odd})$ $\qquad$ $\mathcal{N}_{|\Sigma} = (\mathbb{N}, +, 1, \leq)$

Combining theories

Syntactic view: $\mathcal{T}_1 + \mathcal{T}_2 = \mathcal{T}_1 \cup \mathcal{T}_2 \subseteq F_{\Sigma_1 \cup \Sigma_2}(X)$

$\mathrm{Mod}(\mathcal{T}_1 \cup \mathcal{T}_2) = \{\mathcal{A} \in (\boldsymbol{\Sigma}_1 \cup \boldsymbol{\Sigma}_2)\text{-alg} \mid \mathcal{A} \models G, \text{ for all } G \text{ in } \mathcal{T}_1 \cup \mathcal{T}_2\}$

Semantic view: Let $\mathcal{M}_i = \mathrm{Mod}(\mathcal{T}_i), i = 1, 2$

$\mathcal{M}_1 + \mathcal{M}_2 = \{\mathcal{A} \in (\boldsymbol{\Sigma}_1 \cup \boldsymbol{\Sigma}_2)\text{-alg} \mid \mathcal{A}_{|\Sigma_i} \in \mathcal{M}_i \text{ for } i = 1, 2\}$

$$
\begin{aligned}
\mathcal{A} \in \mathrm{Mod}(\mathcal{T}_1 \cup \mathcal{T}_2) \quad & \text{iff} \quad \mathcal{A} \models G, \text{ for all } G \text{ in } \mathcal{T}_1 \cup \mathcal{T}_2 \\
& \text{iff} \quad \mathcal{A}_{|\Sigma_i} \models G, \text{ for all } G \text{ in } \mathcal{T}_i, i = 1, 2 \\
& \text{iff} \quad \mathcal{A}_{|\Sigma_i} \in \mathcal{M}_i, i = 1, 2 \\
& \text{iff} \quad \mathcal{A} \in \mathcal{M}_1 + \mathcal{M}_2
\end{aligned}
$$

Example

1. Presburger arithmetic + UIF

$\mathsf{Th}(\mathbb{Z}_+) \cup \mathit{UIF}$ $\quad\quad\quad \Sigma = (\Omega, \Pi)$

Models: $(A, 0, s, +, \{f_A\}_{f \in \Omega}, \leq, \{P_A\}_{P \in \Pi})$

where $(A, 0, s, +, \leq) \in \mathsf{Mod}(\mathsf{Th}(\mathbb{Z}_+))$.

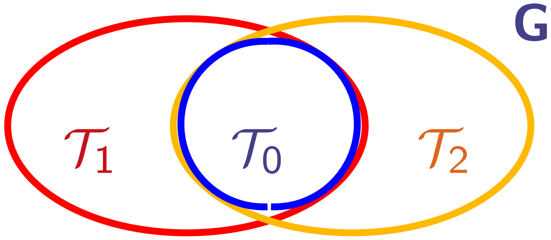
Combinations of theories

The combined decidability problem

For $i = 1, 2$
- let $\mathcal{T}_i$ be a first-order theory in signature Σ_i
- assume the $\mathcal{T}_i$ ground satisfiability problem is decidable

Question:

Is the ground satisfiability problem for $\mathcal{T}_1 + \mathcal{T}_2$ decidable?



Goal: Modular Reasoning

$\mathcal{T}_0$: Σ_0-theory.

$\mathcal{T}_i$: Σ_i-theory; $\mathcal{T}_0 \subseteq \mathcal{T}_i$ $\Sigma_0 \subseteq \Sigma_i$.

Example:

lists($\mathbb{R}$) $\cup$ **arrays**($\mathbb{R}$)

Can use provers for $\mathcal{T}_1, \mathcal{T}_2$ as blackboxes to prove theorems in $\mathcal{T}_1 \cup \mathcal{T}_2$?

Which information needs to be exchanged between the provers?

Combination of theories over disjoint signatures

The Nelson/Oppen procedure

Given: $\mathcal{T}_1, \mathcal{T}_2$ first-order theories with signatures Σ_1, Σ_2

Assume that $\Sigma_1 \cap \Sigma_2 = \emptyset$ (share only $\approx$)

P_i decision procedures for satisfiability of ground formulae w.r.t. $\mathcal{T}_i$

ϕ quantifier-free formula over $\Sigma_1 \cup \Sigma_2$

Task: Check whether ϕ is satisfiable w.r.t. $\mathcal{T}_1 \cup \mathcal{T}_2$

Note: Restrict to conjunctive quantifier-free formulae

$\phi \mapsto DNF(\phi)$

$DNF(\phi)$ satisfiable in $\mathcal{T}$ iff one of the disjuncts satisfiable in $\mathcal{T}$

Example

[Nelson & Oppen, 1979]

Theories

$\mathcal{R}$	theory of rationals	$\Sigma_{\mathcal{R}} = \{\leq, +, -, 0, 1\}$	$\approx$
$\mathcal{L}$	theory of lists	$\Sigma_{\mathcal{L}} = \{\text{car}, \text{cdr}, \text{cons}\}$	$\approx$
$\mathcal{E}$	theory of equality (UIF)	Σ: free function and predicate symbols	$\approx$

Problems:

1. $\mathcal{R} \cup \mathcal{L} \cup \mathcal{E} \models \forall x, y (x \leq y \wedge y \leq x + \text{car}(\text{cons}(0, x)) \wedge P(h(x) - h(y)) \rightarrow P(0))$

2. Is the following conjunction:

$$c \leq d \ \wedge \ d \leq c + \text{car}(\text{cons}(0, c)) \ \wedge \ P(h(c) - h(d)) \ \wedge \ \neg P(0)$$

satisfiable in $\mathcal{R} \cup \mathcal{L} \cup \mathcal{E}$?

An Example

	$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
Σ	$\{\leq, +, -, 0, 1\}$	$\{\text{car}, \text{cdr}, \text{cons}\}$	$F \cup P$
Axioms	$x + 0 \approx x$	$\text{car}(\text{cons}(x, y)) \approx x$	
	$x - x \approx 0$	$\text{cdr}(\text{cons}(x, y)) \approx y$	
(univ.	$+$ is A, C	$\text{at}(x) \vee \text{cons}(\text{car}(x), \text{cdr}(x)) \approx x$	
quantif.)	$\leq$ is R, T, A	$\neg\text{at}(\text{cons}(x, y))$	
	$x \leq y \vee y \leq x$		
	$x \leq y \rightarrow x + z \leq y + z$		

Is the following conjunction:

$$c \leq d \;\wedge\; d \leq c + \text{car}(\text{cons}(0, c)) \;\wedge\; P(h(c) - h(d)) \;\wedge\; \neg P(0)$$

satisfiable in $\mathcal{R} \cup \mathcal{L} \cup \mathcal{E}$?

Step 1: Purification

Given: ϕ conjunctive quantifier-free formula over $\Sigma_1 \cup \Sigma_2$

Task: Find ϕ_1, ϕ_2 s.t. ϕ_i is a pure Σ_i-formula and $\phi_1 \wedge \phi_2$ equivalent with ϕ

$$
\begin{array}{lcl}
f(s_1, \ldots, s_n) \approx g(t_1, \ldots, t_m) & \mapsto & u \approx f(s_1, \ldots, s_n) \wedge u \approx g(t_1, \ldots, t_m) \\
f(s_1, \ldots, s_n) \not\approx g(t_1, \ldots, t_m) & \mapsto & u \approx f(s_1, \ldots, s_n) \wedge v \approx g(t_1, \ldots, t_m) \wedge u \not\approx v \\
(\neg)P(\ldots, s_i, \ldots) & \mapsto & (\neg)P(\ldots, u, \ldots) \wedge u \approx s_i \\
(\neg)P(\ldots, s_i[t], \ldots) & \mapsto & (\neg)P(\ldots, s_i[t \mapsto u], \ldots) \wedge u \approx t \\
\quad \text{where } t \approx f(t_1, \ldots, t_n) & &
\end{array}
$$

Termination: Obvious

Correctness: $\phi_1 \wedge \phi_2$ and ϕ equisatisfiable.

Step 1: Purification

$$c \leq d \;\wedge\; d \leq c + \mathrm{car}(\mathrm{cons}(0, c)) \;\wedge\; P(h(c) - h(d)) \;\wedge\; \neg P(0)$$

Step 1: Purification

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathsf{car}(\mathsf{cons}(0, c))}_{c_1} \;\wedge\; P(h(c) - h(d)) \;\wedge\; \neg P(0)$$

Step 1: Purification

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathsf{car}(\mathsf{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{h(c) - h(d)}_{c_2}) \;\wedge\; \neg P(0)$$

Step 1: Purification

$$c \leq d \ \wedge\ d \leq c + \underbrace{\mathrm{car}(\mathrm{cons}(0, c))}_{c_1} \ \wedge\ P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \ \wedge\ \neg P(\underbrace{0}_{c_5})$$

Step 1: Purification

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathsf{car}(\mathsf{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{0}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \mathsf{car}(\mathsf{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$

Step 1: Purification

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathrm{car}(\mathrm{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{0}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \mathrm{car}(\mathrm{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$
satisfiable	satisfiable	satisfiable

Step 2: Propagation

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathrm{car}(\mathrm{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{0}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \mathrm{car}(\mathrm{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$

deduce and propagate equalities between constants entailed by components

Step 2: Propagation

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathrm{car}(\mathrm{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{\;0\;}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \mathrm{car}(\mathrm{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$

$$c_1 \approx c_5$$

Step 2: Propagation

$$\underbrace{c \leq d}\ \wedge\ d \leq c + \underbrace{\text{car}(\text{cons}(0, c))}_{c_1}\ \wedge\ P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2})\ \wedge\ \neg P(\underbrace{0}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \text{car}(\text{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$
$c_1 \approx c_5$	$c_1 \approx c_5$	
$c \approx d$		

Step 2: Propagation

$$c \leq d \;\wedge\; d \leq c + \underbrace{\mathsf{car}(\mathsf{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{0}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \leq d$	$c_1 \approx \mathsf{car}(\mathsf{cons}(c_5, c))$	$P(c_2)$
$d \leq c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$
$c_1 \approx c_5$	$c_1 \approx c_5$	$c \approx d$
$c \approx d$		$c_3 \approx c_4$

Step 2: Propagation

$$c \le d \;\wedge\; d \le c + \underbrace{\mathrm{car}(\mathrm{cons}(0, c))}_{c_1} \;\wedge\; P(\underbrace{\underbrace{h(c)}_{c_3} - \underbrace{h(d)}_{c_4}}_{c_2}) \;\wedge\; \neg P(\underbrace{\;0\;}_{c_5})$$

$\mathcal{R}$	$\mathcal{L}$	$\mathcal{E}$
$c \le d$	$c_1 \approx \mathrm{car}(\mathrm{cons}(c_5, c))$	$P(c_2)$
$d \le c + c_1$		$\neg P(c_5)$
$c_2 \approx c_3 - c_4$		$c_3 \approx h(c)$
$c_5 \approx 0$		$c_4 \approx h(d)$
$c_1 \approx c_5$	$c_1 \approx c_5$	$c \approx d$
$c \approx d$		$c_3 \approx c_4$
$c_2 \approx c_5$		$\bot$

The Nelson-Oppen algorithm

ϕ conjunction of literals

Step 1. Purification $\mathcal{T}_1 \cup \mathcal{T}_2 \cup \phi \mapsto (\mathcal{T}_1 \cup \phi_1) \cup (\mathcal{T}_2 \cup \phi_2)$:

where ϕ_i is a pure Σ_i-formula and $\phi_1 \wedge \phi_2$ is equisatisfiable with ϕ.

Step 2. Propagation.

The decision procedure for ground satisfiability for $\mathcal{T}_1$ and $\mathcal{T}_2$ fairly exchange information concerning entailed unsatisfiability of constraints in the shared signature i.e. clauses over the shared variables.

until an inconsistency is detected or a saturation state is reached.

The Nelson-Oppen algorithm

ϕ conjunction of literals

Step 1. Purification $\mathcal{T}_1 \cup \mathcal{T}_2 \cup \phi \mapsto (\mathcal{T}_1 \cup \phi_1) \cup (\mathcal{T}_2 \cup \phi_2)$:

where ϕ_i is a pure Σ_i-formula and $\phi_1 \wedge \phi_2$ is equisatisfiable with ϕ.

> not problematic; requires linear time

Step 2. Propagation.

The decision procedure for ground satisfiability for $\mathcal{T}_1$ and $\mathcal{T}_2$ fairly exchange information concerning entailed unsatisfiability of constraints in the shared signature
i.e. clauses over the shared variables.

until an inconsistency is detected or a saturation state is reached.

> not problematic; termination guaranteed
> Sound: if inconsistency detected input unsatisfiable
> Complete: under additional assumptions

The Nelson-Oppen algorithm

Termination: only finitely many shared variables to be identified

Soundness: If procedure answers "unsatisfiable" then ϕ is unsatisfiable

Completeness: Under additional hypotheses

Consider stably infinite theories.

$\mathcal{T}$ is stably infinite iff for every quantifier-free formula ϕ ϕ satisfiable in $\mathcal{T}$ iff ϕ satisfiable in an infinite model of $\mathcal{T}$.

Note: This restriction is not mentioned in [Nelson Oppen 1979];

introduced by Oppen in 1980.

With this additional condition completeness can be proved.

Applications

1. Decision Procedures for data types

- **A decidable fragment of the theory of arrays**

  $\mapsto$ reduction to reasoning in the combination of Presburger arithmetic and uninterpreted function symbols

  Tim Taubitz

- **A decidable fragment of the theory of pointer structures**

  $\mapsto$ reduction to reasoning in the combination of the theory uninterpreted function symbols and the ßcalartheories.

  Jouliet Mesto

Applications

2. Program Verification

Program	$\mapsto$	$T = (\Sigma, \mathsf{Init}, \mathsf{Update}(\Sigma, \Sigma'))$
Safety Property	$\mapsto$	Formula Safe

Task: Prove that the safety property always holds (in general difficult)

Invariant checking

$\mathsf{Init} \models \mathsf{Safe}$

$\mathsf{Safe} \wedge \mathsf{Update}(\Sigma, \Sigma') \models \mathsf{Safe}'$

Bounded model checking: given $k \in \mathbb{N}$. Prove that for all $n \leq k$:

$\mathsf{Init}(\Sigma^0) \wedge \mathsf{Update}|(\Sigma^0, \Sigma^1) \wedge \cdots \wedge \mathsf{Update}|(\Sigma^{n-1}, \Sigma^n) \models \mathsf{Safe}(\Sigma^n)$

Summary

- Logical Theories
- Decidability/Undecidability
- Combination of Logical Theories

  The Nelson/Oppen Method for reasoning in

  combinations of theories with disjoint signatures
- Applications