Decision Procedures for Verification

Decision Procedures (1)

16.12.2014

Viorica Sofronie-Stokkermans

sofronie@uni-koblenz.de

Exam

Several possibilities:

Friday, 27.02.2015

Thursday, 12.03.2015

Friday, 13.03.2015

Chosen:

Thursday, 12.03.2015, 13:00-15:00

Until now:

Syntax (one-sorted signatures vs. many-sorted signatures)

Semantics

Structures (also many-sorted)

Models, Validity, and Satisfiability

Entailment and Equivalence

Theories (Syntactic vs. Semantics view)

Algorithmic Problems: Check satisfiability

Until now:

Normal Forms

Herbrand Models

Resolution

– Soundness, refutational completeness, refinements

– Consequences: Compactness of FOL; The Löwenheim-Skolem Theorem;
Craig interpolation

Decidable subclasses of FOL

The Bernays-Schönfinkel class
(definition; decidability;tractable fragment: Horn clauses)

The Ackermann class

Today

The monadic class

Decision procedures

Congruence closure

The Monadic Class

Monadic first-order logic (MFO) is FOL (without equality) over purely relational signatures $\Sigma = (\Omega, \Pi)$, where $\Omega = \emptyset$, and every $p \in \Pi$ has arity 1.

Abstract syntax:

$\Phi := \top \mid P(x) \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \forall x \Phi \quad \mid \Phi_1 \vee \Phi_2 \mid \Phi_1 \rightarrow \Phi_2 \mid \Phi_1 \leftrightarrow \Phi_2 \mid \exists \Phi$

Idea. Let Φ be a MFO formula with k predicate symbols.

Let $\mathcal{A} = (U_{\mathcal{A}}, \{p_{\mathcal{A}}\}_{p \in \Pi})$ be a Σ-algebra. The only way to distinguish the elements of $U_{\mathcal{A}}$ is by the atomic formulae $p(x)$, $p \in \Pi$.

- the elements which $a \in U_{\mathcal{A}}$ which belong to the same $p_{\mathcal{A}}$'s, $p \in \Pi$ can be collapsed into one single element.
- if $\Pi = \{p^1, \ldots, p^k\}$ then what remains is a *finite structure* with at most 2^k elements.
- the truth value of a formula: computed by evaluating all subformulae.

The Monadic Class

MFO Abstract syntax: $\Phi := \top \mid P(x) \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \forall x \Phi$

Theorem (Finite model theorem for MFO). If Φ is a satisfiable MFO formula with k predicate symbols then Φ has a model where the domain is a subset of $\{0,1\}^k$.

Proof: Let $\mathcal{B} = (\{0,1\}^k, \{p^1_{\mathcal{B}}, \dots, p^k_{\mathcal{B}}\})$, where $p^i_{\mathcal{B}}{=}\{(b_1, \dots, b_k) \mid b_i{=}1\}$.

Let $\mathcal{A} = (U_{\mathcal{A}}, \{p^1_{\mathcal{A}}, \dots, p^k_{\mathcal{A}}\})$, $\beta : X \to U_{\mathcal{A}}$ be such that $(\mathcal{A}, \beta) \models \Phi$.
We construct a model for Φ with cardinality at most 2^k as follows:

- Let $h : \mathcal{A} \to \mathcal{B}$ be defined for all $a \in U_{\mathcal{A}}$ by:

  $$h(a) = (b_1, \dots, b_k) \text{ where } b_i = 1 \text{ if } a \in p^i_{\mathcal{A}} \text{ and } 0 \text{ otherwise.}$$

  Then $a \in p^i_{\mathcal{A}}$ iff $h(a) \in p^i_{\mathcal{B}}$ for all $a \in U_{\mathcal{A}}$ and all $i = 1, \dots, k$.
- Let $\mathcal{B}' = (\{0,1\}^k \cap h(U_{\mathcal{A}}), \{p^1_{\mathcal{B}} \cap h(U_{\mathcal{A}}), \dots, p^k_{\mathcal{B}} \cap h(U_{\mathcal{A}})\})$.
- We show that $(\mathcal{B}', \beta \circ h) \models \Phi$.

The Monadic Class

MFO Abstract syntax: $\Phi := \top \mid P(x) \mid \Phi_1 \wedge \Phi_2 \mid \neg\Phi \mid \forall x \Phi$

Theorem (Finite model theorem for MFO). If Φ is a satisfiable MFO formula with k predicate symbols then Φ has a model where the domain is a subset of $\{0,1\}^k$.

Proof: Let $\mathcal{B} = (\{0,1\}^k, \{p^1_\mathcal{B}, \dots, p^k_\mathcal{B}\})$, where $p^i_\mathcal{B}$=$\{(b_1, \dots, b_k) \mid b_i$=$1\}$.

Let $\mathcal{A} = (U_\mathcal{A}, \{p^1_\mathcal{A}, \dots, p^k_\mathcal{A}\})$, $\beta : X \to U_\mathcal{A}$ be such that $(\mathcal{A}, \beta) \models \Phi$.
We construct a model for Φ with cardinality at most 2^k as follows:

- Let $h : \mathcal{A} \to \mathcal{B}$ be defined for all $a \in U_\mathcal{A}$ by:

  $$h(a) = (b_1, \dots, b_k) \text{ where } b_i = 1 \text{ if } a \in p^i_\mathcal{A} \text{ and 0 otherwise.}$$

  Then $a \in p^i_\mathcal{A}$ iff $h(a) \in p^i_\mathcal{B}$ for all $a \in U_\mathcal{A}$ and all $i = 1, \dots, k$.
- Let $\mathcal{B}' = (\{0,1\}^k \cap h(U_\mathcal{A}), \{p^1_\mathcal{B} \cap h(U_\mathcal{A}), \dots, p^k_\mathcal{B} \cap h(U_\mathcal{A})\})$.
- We show that $(\mathcal{B}', \beta \circ h)(\Phi) = \mathcal{A}(\beta)(\Phi)$. Structural induction

The Monadic Class

To show:
$(\mathcal{A}(\beta)(\Phi) = \mathcal{B}'(\beta \circ h)(\Phi)$.

Induction on the structure of Φ

Induction base: Show that claim is true for all atomic formulae

- $\Phi = \top$ OK

- $\Phi = p^i(x)$.

  Then the following are equivalent:

  (1) $(\mathcal{A}, \beta) \models \Phi$
  (2) $\beta(x) \in p^i_{\mathcal{A}}$ (definition)
  (3) $h(\beta(x)) \in p^i_{\mathcal{B}}$ (definition of h and of $p^i_{\mathcal{B}}$)
  (4) $(\mathcal{B}', \beta \circ h) \models \Phi$ (definition)

The Monadic Class

Induction on the structure of Φ

Let Φ be a formula which is not atomic.

Assume statement holds for the (direct) subformulae of Φ. Prove that it holds for Φ.

- $\Phi = \Phi_1 \wedge \Phi_2$

  Assume $(\mathcal{A}, \beta) \models \Phi$. Then $(\mathcal{A}, \beta) \models \Phi_i$, $i = 1, 2$.

  By induction hypothesis, $(\mathcal{B}', \beta \circ h) \models \Phi_i$, $i = 1, 2$.

  Thus, $(\mathcal{B}', \beta \circ h) \models \Phi = \Phi_1 \wedge \Phi_2$

  The converse can be proved similarly.

- $\Phi = \neg\Phi_1$

  The following are equivalent:

  (1) $(\mathcal{A}, \beta) \models \Phi = \neg\Phi_1$.

  (2) $\mathcal{A}(\beta)(\Phi_1) = 0$

  (3) $\mathcal{B}'(\beta \circ h)(\Phi_1) = 0$ (induction hypothesis)

  (4) $(\mathcal{B}', \beta \circ h) \models \Phi = \neg\Phi_1$

The Monadic Class

- $\Phi = \forall x \Phi_1(x)$.

  Then the following are equivalent:

  (1) $(\mathcal{A}, \beta) \models \Phi$

  (2) $\mathcal{A}(\beta[x \mapsto a])(\Phi_1) = 1$ for all $a \in U_{\mathcal{A}}$

  (3) $\mathcal{B}'(\beta[x \mapsto a] \circ h)(\Phi_1) = 1$ for all $a \in U_{\mathcal{A}}$ (ind. hyp)

  (4) $\mathcal{B}'(\beta \circ h[x \mapsto b])(\Phi_1) = 1$ for all $b \in \{0, 1\}^k \cap h(A)$

  (5) $(\mathcal{B}', \beta \circ h) \models \Phi$

The Monadic Class

Resolution-based decision procedure for the Monadic Class (and for several other classes):

William H. Joyner Jr.
Resolution Strategies as Decision Procedures.
J. ACM 23(3): 398-417 (1976)

Idea:

- Use orderings to restrict the possible inferences
- Identify a class of clauses (with terms of bounded depth) which contains the type of clauses generated from the respective fragment and is closed under ordered resolution (+ red. elim. criteria)
- Show that a saturation of the clauses can be obtained in finite time

The Monadic Class

Resolution-based decision procedure for the Monadic Class:

$$\begin{array}{ll} \Phi : & \forall \overline{x}_1 \exists \overline{y}_1 \ldots \forall \overline{x}_k \exists \overline{y}_k (....p^s(x_i)......p^l(y_i)...) \\ \mapsto & \forall \overline{x}_1 \ldots \forall \overline{x}_k (...p^s(x_i)...p^l(f_{\mathsf{sk}}(\overline{x}_1, \ldots, \overline{x}_i)...) \end{array}$$

Consider the class MON of clauses with the following properties:

- no literal of heigth greater than 2 appears
- each variable-disjoint partition has at most $n = \sum_{i=1} |\overline{x}_i|$ variables (can order the variables as $x_1, \ldots, x_n$)
- the variables of each non-ground block can occur either in atoms $p(x_i)$ or in atoms $P(f_{\mathsf{sk}}(x_1, \ldots, x_t))$, $0 \leq t \leq n$

It can be shown that this class contains all CNF's of formulae in the monadic class and is closed under ordered resolution.

3.2 Deduction problems

Satisfiability w.r.t. a theory

Satisfiability w.r.t. a theory

Example

Let $\Sigma = (\{e/0, */2, i/1\}, \emptyset)$

Let $\mathcal{F}$ consist of all (universally quantified) group axioms:

$$\begin{array}{ll} \forall x, y, z & x * (y * z) \approx (x * y) * z \\ \forall x & x * i(x) \approx e \quad \wedge \quad i(x) * x \approx e \\ \forall x & x * e \approx x \quad \wedge \quad e * x \approx x \end{array}$$

Question: Is $\forall x, y (x * y = y * x)$ entailed by $\mathcal{F}$?

Satisfiability w.r.t. a theory

Example

Let $\Sigma = (\{e/0, */2, i/1\}, \emptyset)$

Let $\mathcal{F}$ consist of all (universally quantified) group axioms:

$$\begin{array}{ll} \forall x, y, z & x * (y * z) \approx (x * y) * z \\ \forall x & x * i(x) \approx e \quad \wedge \quad i(x) * x \approx e \\ \forall x & x * e \approx x \quad \wedge \quad e * x \approx x \end{array}$$

Question: Is $\forall x, y(x * y = y * x)$ entailed by $\mathcal{F}$?

Alternative question:

Is $\forall x, y(x * y = y * x)$ true in the class of all groups?

Logical theories

Syntactic view

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

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

Semantic view

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

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

Decidable theories

Let $\Sigma = (\Omega, \Pi)$ be a signature.

$\mathcal{M}$: class of Σ-algebras. $\mathcal{T} = \mathrm{Th}(\mathcal{M})$ 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 $\mathcal{T} = \mathrm{Th}(\mathrm{Mod}(\mathcal{F}))$ 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

- Th$((\mathbb{Z}, \{0, 1, +, *\}, \{\leq\}))$
- Peano arithmetic
- Th(Σ-alg)

Peano arithmetic

Peano axioms:

$\forall x\, \neg(x + 1 \approx 0)$	(zero)
$\forall x \forall y\, (x + 1 \approx y + 1 \to x \approx y$	(successor)
$F[0] \wedge (\forall x\, (F[x] \to F[x + 1]) \to \forall x F[x])$	(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 0)
$\forall x, y\, (x * (y + 1) \approx x * y + x)$	(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 Goedel's incompleteness theorem)

Examples

Undecidable theories

- $\bullet$Th$((\mathbb{Z}, \{0, 1, +, *\}, \{\leq\}))$
- $\bullet$ Peano arithmetic
- $\bullet$Th(Σ-alg)

Idea of undecidability proof: Suppose there is an algorithm P that, given a formula in one of the theories above decides whether that formula is valid.

We use P to give a decision algorithm for the language

$\{(G(M), w) | G(M)$ is the Gödelisation of a TM M that accepts the string w $\}$

As the latter problem is undecidable, this will show that P cannot exist.

Examples

Undecidable theories

•Th(($\mathbb{Z}$, $\{0, 1, +, *\}$, $\{\leq\}$))
• Peano arithmetic

•Th(Σ-alg)

Idea of undecidability proof: (ctd)

(1) For Th(($\mathbb{Z}$, $\{0, 1, +, *\}$, $\{\leq\}$)) and Peano arithmetic:

multiplication can be used for modeling Gödelisation

(2) For Th(Σ-alg):

Given M and w, we create a FOL signature and a set of formulae over this signature encoding the way M functions, and a formula which is valid iff M accepts w.

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) }

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

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

Problems

$\mathcal{T}$: first-order theory in signature Σ; $\mathcal{L}$ class of (closed) Σ-formulae

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

Common restrictions on $\mathcal{L}$

	$\mathsf{Pred} = \emptyset$	$\{\phi \in \mathcal{L} \mid \mathcal{T} \models \phi\}$
$\mathcal{L}=\{\forall x A(x) \mid A \text{ atomic}\}$	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	$\mathsf{Th}_{\forall}$
$\mathcal{L}=\{\exists x A_1 \wedge \ldots \wedge A_n \mid A_i \text{ atomic}\}$	unification problem	$\mathsf{Th}_{\exists}$
$\mathcal{L}=\{\forall x \exists x A_1 \wedge \ldots \wedge A_n \mid A_i \text{ atomic}\}$	unification with constants	$\mathsf{Th}_{\forall\exists}$

$\mathcal{T}$-validity vs. $\mathcal{T}$-satisfiability

$\mathcal{T}$-validity: Let $\mathcal{T}$ be a first-order theory in signature Σ
Let $\mathcal{L}$ be a class of (closed) Σ-formulae

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

Remark: $\mathcal{T} \models \phi$ iff $\mathcal{T} \cup \neg\phi$ unsatisfiable

Every $\mathcal{T}$-validity problem has a dual $\mathcal{T}$-satisfiability problem:

$\mathcal{T}$-satisfiability: Let $\mathcal{T}$ be a first-order theory in signature Σ
Let $\mathcal{L}$ be a class of (closed) Σ-formulae

$$\neg\mathcal{L} = \{\neg\phi \mid \phi \in \mathcal{L}\}$$

Given ψ in $\neg\mathcal{L}$, is it the case that $\mathcal{T} \cup \psi$ is satisfiable?

$\mathcal{T}$-validity vs. $\mathcal{T}$-satisfiability

Common restrictions on $\mathcal{L}$ / $\neg\mathcal{L}$

$\mathcal{L}$	$\neg\mathcal{L}$
$\{\forall x A(x) \mid A \text{ atomic}\}$	$\{\exists x \neg A(x) \mid A \text{ atomic}\}$
$\{\forall x (A_1 \wedge \ldots \wedge A_n \rightarrow B) \mid A_i, B \text{ atomic}\}$	$\{\exists x (A_1 \wedge \ldots \wedge A_n \wedge \neg B) \mid A_i, B \text{ atomic}\}$
$\{\forall x \bigvee L_i \mid L_i \text{ literals}\}$	$\{\exists x \bigwedge L'_i \mid L'_i \text{ literals}\}$
$\{\forall x \phi(x) \mid \phi(x) \text{ unquantified}\}$	$\{\exists x \phi'(x) \mid \phi'(x) \text{ unquantified}\}$

validity problem for universal formulae ground satisfiability problem

$\mathcal{T}$-validity vs. $\mathcal{T}$-satisfiability

Common restrictions on $\mathcal{L}$ / $\neg\mathcal{L}$

$\mathcal{L}$	$\neg\mathcal{L}$
$\{\forall x A(x) \mid A \text{ atomic}\}$	$\{\exists x \neg A(x) \mid A \text{ atomic}\}$
$\{\forall x (A_1 \wedge \ldots \wedge A_n \rightarrow B) \mid A_i, B \text{ atomic}\}$	$\{\exists x (A_1 \wedge \ldots \wedge A_n \wedge \neg B) \mid A_i, B \text{ atomic}\}$
$\{\forall x \bigvee L_i \mid L_i \text{ literals}\}$	$\{\exists x \bigwedge L'_i \mid L'_i \text{ literals}\}$
$\{\forall x \phi(x) \mid \phi(x) \text{ unquantified}\}$	$\{\exists x \phi'(x) \mid \phi'(x) \text{ unquantified}\}$

validity problem for universal formulae | ground satisfiability problem

In what follows we will focus on the problem of checking the satisfiability of conjunctions of ground literals

$\mathcal{T}$-validity vs. $\mathcal{T}$-satisfiability

$$
\begin{array}{lll}
\mathcal{T} \models \forall x A(x) & \text{iff} & \mathcal{T} \cup \exists x \neg A(x) \text{ unsatisfiable} \\
\mathcal{T} \models \forall x (A_1 \wedge \cdots \wedge A_n \rightarrow B) & \text{iff} & \mathcal{T} \cup \exists x (A_1 \wedge \cdots \wedge A_n \wedge \neg B) \text{ unsatisfiable} \\
\mathcal{T} \models \forall x (\bigvee_{i=1}^{n} A_i \vee \bigvee_{j=1}^{m} \neg B_j) & \text{iff} & \mathcal{T} \cup \exists x (\neg A_1 \wedge \cdots \wedge \neg A_n \wedge B_1 \wedge \cdots \wedge B_m) \\
& & \text{unsatisfiable}
\end{array}
$$

$\mathcal{T}$-satisfiability vs. Constraint Solving

The field of Constraint Solving also deals with satisfiability problems

But be careful:

- in Constraint Solving one is interested if a formula is satisfiable in a given, fixed model of $\mathcal{T}$.
- in $\mathcal{T}$-satisfiability one is interested if a formula is satisfiable in any model of $\mathcal{T}$ at all.

3.3. Theory of Uninterpreted Function Symbols

Why?

- Reasoning about equalities is important in automated reasoning

- Applications to program verification
  (approximation: abstract from additional properties)

Application: Compiler Validation

Example: prove equivalence of source and target program

```
1:  y := 1
2:  if z = x*x*x
3:    then y := x*x + y
4:  endif
```

```
1: y := 1
2: R1 := x*x
3: R2 := R1*x
4: jmpNE(z,R2,6)
5: y := R1+1
```

To prove: (indexes refer to values at line numbers)

$$
\begin{gathered}
y_1 \approx 1 \wedge [(z_0 \approx x_0 * x_0 * x_0 \wedge y_3 \approx x_0 * x_0 + y_1) \vee (z_0 \not\approx x_0 * x_0 * x_0 \wedge y_3 \approx y_1)] \wedge \\
y'_1 \approx 1 \wedge R1_2 \approx x'_0 * x'_0 \wedge R2_3 \approx R1_2 * x'_0 \wedge \\
\wedge [(z'_0 \approx R2_3 \wedge y'_5 \approx R1_2 + 1) \vee (z'_0 \neq R2_3 \wedge y'_5 \approx y'_1)] \wedge \\
x_0 \approx x'_0 \wedge y_0 \approx y'_0 \wedge z_0 \approx z'_0 \implies x_0 \approx x'_0 \wedge y_3 \approx y'_5 \wedge z_0 \approx z'_0
\end{gathered}
$$

Possibilities for checking it

(1) **Abstraction.**
Consider $*$ to be a "free" function symbol (forget its properties). Test it property can be proved in this approximation. If so, then we know that implication holds also under the normal interpretation of $*$.

(2) **Reasoning about formulae in fragments of arithmetic.**

Uninterpreted function symbols

Let $\Sigma = (\Omega, \Pi)$ be arbitrary

Let $\mathcal{M} = \Sigma$-alg be 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-algebras.

in general undecidable

Decidable fragment:

e.g. the class $\mathrm{Th}_{\forall}(\Sigma\text{-alg})$ of all universal formulae which are true in all Σ-algebras.

Uninterpreted function symbols

Assume $\Pi = \emptyset$ (and $\approx$ is the only predicate)

In this case we denote the theory of uninterpreted function symbols by $UIF(\Sigma)$ (or UIF when the signature is clear from the context).

This theory is sometimes called the theory of free functions and denoted Free(Σ)

Uninterpreted function symbols

Theorem 3.3.1

The following are equivalent:

(1) testing validity of universal formulae w.r.t. UIF is decidable

(2) testing validity of (universally quantified) clauses w.r.t. UIF is decidable

Proof: Follows from the fact that any universal formula is equivalent to a conjunction of (universally quantified) clauses.

Solution 1

Task:

Check if $UIF \models \forall \overline{x}(s_1(\overline{x}) \approx t_1(\overline{x}) \wedge \cdots \wedge s_k(\overline{x}) \approx t_k(\overline{x}) \rightarrow \bigvee_{j=1}^{m} s'_j(\overline{x}) \approx t'_j t(\overline{x}))$

Solution 1:

The following are equivalent:

(1) $(\bigwedge_i s_i \approx t_i) \rightarrow \bigvee_j s'_j \approx t'_j$ is valid

(2) $Eq(\sim) \wedge \mathrm{Con}(f) \wedge (\bigwedge_i s_i \sim t_i) \wedge (\bigwedge_j s'_j \not\sim t'_j)$ is unsatisfiable.

where $Eq(\sim) : \mathrm{Refl}(\sim) \wedge \mathrm{Sim}(\sim) \wedge \mathrm{Trans}(\sim)$

$\mathrm{Con}(f) : \forall x_1, \dots, x_n, y_1, \dots, y_n(\bigwedge x_i \sim y_i \rightarrow f(x_1, \dots, x_n) \sim f(y_1, \dots, y_n))$

Resolution: inferences between transitivity axioms – nontermination

Solution 2

Task:

Check if $UIF \models \forall \overline{x}(s_1(\overline{x}) \approx t_1(\overline{x}) \wedge \dots \wedge s_k(\overline{x}) \approx t_k(\overline{x}) \rightarrow \bigvee_{j=1}^{m} s'_j(\overline{x}) \approx t'_j(\overline{x}))$

Solution 2: Ackermann's reduction.

Flatten the formula (replace, bottom-up, $f(c)$ with a new constant c_f
$\phi \mapsto FLAT(\phi)$

Theorem 3.3.2: The following are equivalent:

(1) $(\bigwedge_i s_i(\overline{c}) \approx t_i(\overline{c})) \wedge \bigwedge_j s'_j(\overline{c}) \not\approx t'_j(\overline{c})$ is satisfiable
(2) $FC \wedge FLAT[(\bigwedge_i s_i(\overline{c}) \approx t_i(\overline{c})) \wedge \bigwedge_j s'_j(\overline{c}) \not\approx t'_j(\overline{c})]$ is satisfiable

where $FC = \{c_1 = d_1, \dots c_n = d_n \rightarrow c_f = d_f \mid$ whenever $f(c_1, \dots, c_n)$ was renamed to c_f
$f(d_1, \dots, d_n)$ was renamed to $d_f\}$

Note: The problem is decidable in PTIME (see next pages)
Problem: Naive handling of transitivity/congruence axiom $\mapsto O(n^3)$
Goal: Give a faster algorithm

Example

The following are equivalent:

(1) $C := f(a, b) \approx a \wedge f(f(a, b), b) \not\approx a$ is satisfiable

(2) $FC \wedge FLAT[C]$ is satisfiable, where:

$FLAT[f(a, b) \approx a \wedge f(f(a, b), b) \not\approx a]$ is computed by introducing new constants renaming terms starting with f and then replacing in C the terms with the constants:

- $FLAT[\underbrace{f(a, b)}_{a_1} \approx a \wedge \underbrace{f(\underbrace{f(a, b)}_{a_1}, b)}_{a_2} \not\approx a] := a_1 \approx a \wedge a_2 \not\approx a$

$$f(a, b) = a_1$$
$$f(a_1, b) = a_2$$

- $FC := (a \approx a_1 \rightarrow a_1 \approx a_2)$

Thus, the following are equivalent:

(1) $C := f(a, b) \approx a \wedge f(f(a, b), b) \not\approx a$ is satisfiable

(2) $\underbrace{(a \approx a_1 \rightarrow a_1 \approx a_2)}_{FC} \wedge \underbrace{a_1 \approx a \wedge a_2 \not\approx a}_{FLAT[C]}$ is satisfiable

Solution 3

Next time

Next lectures

Thursday, January 8, 2015

Tuesday, January 13, 2015