Decision Procedures in Verification

First-Order Logic (5)

20.12.2021

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de
Decidable subclasses of first-order logic
Applications

Use ordered resolution with selection to give a decision procedure for the Ackermann class.
The Ackermann class

\( \Sigma = (\Omega, \Pi) \), \( \Omega \) is a finite set of constants

The Ackermann class consists of all sentences of the form

\[ \exists x_1 \ldots \exists x_n \forall x \exists y_1 \ldots \exists y_m F(x_1, \ldots, x_n, x, y_1, \ldots, y_m) \]

**Idea:** CNF translation:

1. \[ \exists x_1 \ldots \exists x_n \forall x \exists y_1 \ldots \exists y_m F(x_1, \ldots, x_n, x, y_1, \ldots, y_m) \]
2. \[ \Rightarrow S \forall x F(c_1, \ldots, c_n, x, f_1(x), \ldots, f_m(x)) \]
3. \[ \Rightarrow K \forall x \bigwedge \bigvee L_i(c_1, \ldots, c_n, x, f_1(x), \ldots, f_m(x)) \]

\( c_1, \ldots, c_n \) are Skolem constants

\( f_1, \ldots, f_m \) are unary Skolem functions
The Ackermann class

\( \Sigma = (\Omega, \Pi) \), \( \Omega \) is a finite set of constants

The Ackermann class consists of all sentences of the form

\[ \exists x_1 \ldots \exists x_n \forall \exists y_1 \ldots \exists y_m F(x_1, \ldots, x_n, x, y_1, \ldots, y_m) \]

Idea: CNF translation:

\[ \exists x_1 \ldots \exists x_n \forall \exists y_1 \ldots \exists y_m F(x_1, \ldots, x_n, x, y_1, \ldots, y_m) \Rightarrow^* \forall x \bigwedge \bigvee L_i(c_1, \ldots, c_n, x, f_1(x), \ldots, f_m(x)) \]

The clauses are in the following classes:

\( G = G(c_1, \ldots, c_n) \) ground clauses without function symbols
\( V = V(x, c_1, \ldots, c_n) \) clauses with one variable and without function symbols
\( G_f = G(c_1, \ldots, c_n, f_1, \ldots, f_n) \) ground clauses with function symbols
\( V_f = V(x, c_1, \ldots, c_n, f_1(x), \ldots, f_n(x)) \) clauses with a variable & function symbols
The Ackermann class

\[ G = G(c_1, \ldots, c_n) \] ground clauses without function symbols
\[ V = V(x, c_1, \ldots, c_n) \] clauses with one variable and without function symbols
\[ G_f = G(c_1, \ldots, c_n, f_1, \ldots, f_n) \] ground clauses with function symbols
\[ V_f = V(x, c_1, \ldots, c_n, f_1(x), \ldots, f_n(x)) \] clauses with a variable & function symbols

Term ordering

\( f(t) \succ t \); terms containing function symbols larger than those who do not.
\( B \succ A \) iff exists argument \( u \) of \( B \) such that every argument \( t \) of \( A \): \( u \succ t \)

**Ordered resolution:** \( G \cup V \cup G_f \cup V_f \) is closed under ordered resolution.

\[ G, G \mapsto G; \quad G, V \mapsto G; \quad G, G_f \mapsto \text{nothing}; \quad G, V_f \mapsto \text{nothing} \]
\[ V, V \mapsto V \cup G; \quad V, G_f \mapsto G \cup G_f; \quad V, V_f \mapsto G \cup V \cup G_f \cup V_f \]
\[ G_f, G_f \mapsto G_f; \quad G_f, V_f \mapsto G_f \cup G; \quad V_f, V_f \mapsto G \cup V \cup V_f \cup G_f \]

**Observation 1:** \( G \cup V \cup G_f \cup V_f \) finite set of clauses (up to renaming of variables).
The Ackermann class

\[ G = G(c_1, \ldots, c_n) \] ground clauses without function symbols

\[ V = V(x, c_1, \ldots, c_n) \] clauses with one variable and without function symbols

\[ G_f = G(c_1, \ldots, c_n, f_i) \] ground clauses with function symbols

\[ V_f = V(x, c_1, \ldots, c_n, f_1(x), \ldots, f_n(x)) \] clauses with a variable & function symbols

Term ordering

\[ f(t) \succ t; \] terms containing function symbols larger than those who do not.

\[ B \succ A \text{ iff exists argument } u \text{ of } B \text{ such that every argument } t \text{ of } A: \ u \succ t \]

**Ordered resolution:** \( G \cup V \cup G_f \cup V_f \) is closed under ordered resolution.

\[ G, G \mapsto G; \ G, V \mapsto G; \ G, G_f \mapsto \text{nothing}; \ G, V_f \mapsto \text{nothing} \]

\[ V, V \mapsto V \cup G; \ V, G_f \mapsto G \cup G_f; \ V, V_f \mapsto G \cup V \cup G_f \cup V_f \]

\[ G_f, G_f \mapsto G_f; \ G_f, V_f \mapsto G_f \cup G; \ V_f, V_f \mapsto G \cup V \cup V_f \cup G_f \]

**Observation 2:** No clauses with nested function symbols can be generated.
The Ackermann Class

Conclusion:

Resolution (with implicit factorization) will always terminate if the input clauses are in the class defined before.

Resolution can be used as a decision procedure to check the satisfiability of formulae in the Ackermann class.
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:

$$\Phi := \top \mid P(x) \mid \Phi_1 \land \Phi_2 \mid \neg \Phi \mid \Phi_1 \lor \Phi_2 \mid \forall x \Phi \mid \exists x \Phi$$

Idea. Let $\Phi$ be a MFO formula with $k$ predicate symbols.

Let $\mathcal{A} = (U_\mathcal{A}, \{p_\mathcal{A}\}_{p \in \Pi})$ be a $\Sigma$-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: $\Phi := T | P(x) | \Phi_1 \land \Phi_2 | \neg \Phi | \Phi_1 \lor \Phi_2 | \forall x \Phi | \exists x \Phi$

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

Proof: Let $B = (\{0, 1\}^k, \{p_1^B, \ldots, p_k^B\})$, where $p_i^B = \{(b_1, \ldots, b_k) \mid b_i = 1\}$.
Let $\mathcal{A} = (U_\mathcal{A}, \{p_1^\mathcal{A}, \ldots, p_k^\mathcal{A}\})$, $\beta : X \to U_\mathcal{A}$ be such that $(\mathcal{A}, \beta) \models \Phi$. We construct a model for $\Phi$ with cardinality at most $2^k$ as follows:

- Let $h : \mathcal{A} \to B$ be defined for all $a \in U_\mathcal{A}$ by:
  
  $$h(a) = (b_1, \ldots, b_k) \text{ where } b_i = 1 \text{ if } a \in p_i^\mathcal{A} \text{ and } 0 \text{ otherwise.}$$

  Then $a \in p_i^\mathcal{A}$ iff $h(a) \in p_i^B$ for all $a \in U_\mathcal{A}$ and all $i = 1, \ldots, k$.

- Let $B' = (\{0, 1\}^k \cap h(U_\mathcal{A}), \{p_1^1 \cap h(U_\mathcal{A}), \ldots, p_k^k \cap h(U_\mathcal{A})\})$.
- We show that $(B', \beta \circ h) \models \Phi$. 

The Monadic Class

Let $B = (\{0, 1\}^k, \{p_B^1, \ldots, p_B^k\})$, where $p_B^i = \{(b_1, \ldots, b_k) | b_i = 1\}$.

Let $A = (U_A, \{p_A^1, \ldots, p_A^k\})$, $\beta : X \to U_A$ be such that $(A, \beta) \models \Phi$. We construct a model for $\Phi$ with cardinality at most $2^k$ as follows:

- Let $h : A \to B$ be defined for all $a \in U_A$ by:
  \[ h(a) = (b_1, \ldots, b_k) \text{ where } b_i = 1 \text{ if } a \in p_A^i \text{ and } 0 \text{ otherwise.} \]
  Then $a \in p_A^i$ iff $h(a) \in p_B^i$ for all $a \in U_A$ and all $i = 1, \ldots, k$.

- Let $B' = (\{0, 1\}^k \cap h(U_A), \{p_B^1 \cap h(U_A), \ldots, p_B^k \cap h(U_A)\})$.

- We show that $(B', \beta \circ h) \models \Phi$.

Induction on the structure of $\Phi$

- $\Phi = \top$ OK

- $\Phi = p^i(x)$. Then $(A, \beta) \models \Phi$ iff $\beta(x) \in p_A^i$ iff $h(\beta(x)) \in p_B^i$ iff $(B', \beta \circ h) \models \Phi$. 

The Monadic Class

Let $B = (\{0, 1\}^k, \{p^1_B, \ldots, p^k_B\})$, where $p^i_B = \{(b_1, \ldots, b_k) \mid b_i = 1\}$. Let $A = (U_A, \{p^1_A, \ldots, p^k_A\})$, $\beta : X \rightarrow U_A$ be such that $(A, \beta) \models \Phi$. We construct a model for $\Phi$ with cardinality at most $2^k$ as follows:

- Let $h : A \rightarrow B$ be defined for all $a \in U_A$ by:
  
  $$h(a) = (b_1, \ldots, b_k)$$

  where $b_i = 1$ if $a \in p^i_A$ and 0 otherwise.

  Then $a \in p^i_A$ iff $h(a) \in p^i_B$ for all $a \in U_A$ and all $i = 1, \ldots, k$.

- Let $B' = (\{0, 1\}^k \cap h(U_A), \{p^1_B \cap h(U_A), \ldots, p^k_B \cap h(U_A)\})$.

- We show that $(B', \beta \circ h) \models \Phi$.

Induction on the structure of $\Phi$

- $\Phi = \Phi_1 \land \Phi_2$: standard

- $\Phi = \neg \Phi_1$: standard
The Monadic Class

Let $B = (\{0, 1\}^k, \{p^1_B, \ldots, p^k_B\})$, where $p^i_B = \{(b_1, \ldots, b_k) | b_i = 1\}$.

Let $A = (U_A, \{p^1_A, \ldots, p^k_A\})$, $\beta : X \rightarrow U_A$ be such that $(A, \beta) \models \Phi$. We construct a model for $\Phi$ with cardinality at most $2^k$ as follows:

- Let $h : A \rightarrow B$ be defined for all $a \in U_A$ by:
  \[ h(a) = (b_1, \ldots, b_k) \text{ where } b_i = 1 \text{ if } a \in p^i_A \text{ and } 0 \text{ otherwise.} \]

  Then $a \in p^i_A$ iff $h(a) \in p^i_B$ for all $a \in U_A$ and all $i = 1, \ldots, k$.

- Let $B' = (\{0, 1\}^k \cap h(U_A), \{p^1_B \cap h(U_A), \ldots, p^k_B \cap h(U_A)\})$.

- We show that $(B', \beta \circ h) \models \Phi$.

Induction on the structure of $\Phi$

- $\Phi = \forall x \Phi_1(x)$. Then the following are equivalent:
  - $(A, \beta) \models \Phi$ (i.e. $(A, \beta[x \mapsto a]) \models \Phi_1$ for all $a \in U_A$)
  - $(B', \beta[x \mapsto a] \circ h) \models \Phi_1$ for all $a \in U_A$ (ind. hyp)
  - $(B', \beta \circ h[x \mapsto b]) \models \Phi_1$ for all $b \in \{0, 1\}^k \cap h(A)$ (i.e. $(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.

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:

\[ \Phi : \forall x_1 \exists y_1 \ldots \forall x_k \exists y_k (\ldots \ p^s(x_i) \ldots \ p^l(y_i) \ldots) \]
\[ \mapsto \forall x_1 \ldots \forall x_k (\ldots \ p^s(x_i) \ldots \ p^l(f_{sk}(x_1, \ldots, x_i)) \ldots) \]

Consider the class MON of clauses with the following properties:
- no literal of height greater than 2 appears
- each variable-disjoint partition has at most \( n = \sum_{i=1}^{m} \lvert x_i \rvert \) 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 \( P(f_{sk}(x_1, \ldots, x_t)) \), \( 0 \leq t \leq n \)

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