Universität Koblenz-Landau

FB 4 Informatik

Prof. Dr. Viorica Sofronie-Stokkermans **November 15, 2022**

Exercises for “Decision Procedures for Verification”
Exercise sheet 4

Exercise 4.1: (*2 P*)
Let N be the following set of propositional clauses:

(1) $P \vee \neg Q \vee R$

(2) $P \vee \neg T \vee \neg U \vee V$

(3) $P \vee \neg Q \vee T \vee U \vee \neg V$

(4) $\neg P \vee Q$

(5) $R \vee T$

(6) $R \vee \neg U$

(7) $\neg P \vee S \vee \neg U \vee \neg V$

(8) $\neg R \vee S$

(9) $\neg R \vee T$

(10) $\neg S \vee T \vee U \vee \neg V$

(11) $\neg T \vee U$

(12) $\neg S \vee \neg T \vee \neg U \vee V$

(13) $\neg U \vee \neg V$

Use the DPLL method with backjumping to give a model. Use the DPLL inference rules with a reasonable strategy (i.e., use Fail or Backjump if possible, otherwise use Unit Propagate if possible, otherwise use Decide). If you use the Decide rule, use the largest undefined positive literal according to the ordering $P > Q > R > S > T > U > V$. If you use the Backjump rule, determine a suitable backjump clause using the 1UIP method and use the best possible successor state for that backjump clause.

Exercise 4.2: (*5 P*)
Let N be a set of clauses in propositional logic with the property that each clause consists of two literals. Prove that the satisfiability of N can be checked in polynomial time in the size of N.

Hint (way to a possible solution):

- How many clauses consisting of two literals (over a *finite* set of propositional variables $\Pi = \{P_1, \dots, P_n\}$) exist?

- Analyze the form of possible resolution inferences from N.
- Let N be a set of clauses in propositional logic with the property that each clause consists of two literals. Show that
  - If N is satisfiable then we cannot generate from N, using the resolution calculus, both $P \vee P$ and $\neg P \vee \neg P$ for some propositional variable P.
  - If we cannot generate from N, using the resolution calculus, both $P \vee P$ and $\neg P \vee \neg P$ for some propositional variable P then N is satisfiable.
- Show that the number of inferences by resolution from N which yield different clauses is polynomial in the size of N and in the size of Π. Infer that the satisfiability of N can be checked in polynomial time in the size of N.

Exercise 4.3: *(2 P)*
Let $\Sigma = (\Omega, \Pi)$ be a signature, where $\Omega = \{f/3,\ g/1,\ a/0,\ b/0\}$ and $\Pi = \{p/2\}$; let X be the set of variables $\{x, y, z\}$. Which of the following expressions are terms over Σ and X, which are atoms/literals/clauses/formulae (in first-order logic with equality), which are neither?

(a) $\neg p(g(a), f(x, y, g(a)))$

(b) $f(x, x, x) \approx x$

(c) $p(f(x, x, a), x) \vee p(a, b)$

(d) $p(\neg g(x), g(y))$

(e) $\neg p(f(x, y, y))$

(f) $\neg p(f(x, y), y) \vee p(x, y)$

(g) $p(a, b) \wedge p(x, y) \wedge y$

(h) $\exists y(\neg p(f(y, y, y), y))$

(i) $\forall x \forall y(f(p(x, y), x, x) \approx g(x))$

Exercise 4.4: *(2 P)*
Let $\Sigma = (S, \Omega, \Pi)$ be a many-sorted signature, where $S = \{\mathsf{int}, \mathsf{list}\}$, $\Omega = \{\mathsf{cons}, \mathsf{car}, \mathsf{cdr}, \mathsf{nil}, b\}$ and $\Pi = \{p\}$ with the following arities:

$a(\mathsf{cons}) = \mathsf{int}, \mathsf{list} \to \mathsf{list}$ $\quad a(\mathsf{car}) = \mathsf{list} \to \mathsf{int}$ $\quad a(\mathsf{cdr}) = \mathsf{list} \to \mathsf{list}$
$a(\mathsf{nil}) = \to \mathsf{list}$ (i.e. nil is a constant of sort list)
$a(b) = \to \mathsf{int}$ (i.e. b is a constant of sort int)
$a(p) = \mathsf{int}, \mathsf{list}$.

Let X_{int} be the set of variables of sort int containing $\{i, j, k\}$, and let X_{list} be the set of variables of sort list containing $\{x, y, z\}$. Let $X = \{X_{\mathsf{int}}, X_{\mathsf{list}}\}$. Which of the following expressions are terms over Σ and X, which are atoms/literals/clauses/formulae[1], which are neither?

(a) $\mathsf{cons}(\mathsf{cons}(b, \mathsf{nil}), \mathsf{nil})$

(b) $\mathsf{cons}(b, \mathsf{cons}(b, \mathsf{nil}))$

(c) $\neg p(b, \mathsf{cons}(b, \mathsf{cons}(b, \mathsf{nil})))$

(d) $\neg p(\mathsf{cons}(b, \mathsf{nil}), \mathsf{cons}(b, \mathsf{cons}(b, \mathsf{nil})))$

(e) $\mathsf{cons}(b, \mathsf{cons}(b, \mathsf{nil})) \approx_l \mathsf{cons}(\mathsf{cons}(x, b), \mathsf{nil})$

(f) $\mathsf{cons}(i, \mathsf{cons}(b, \mathsf{nil})) \approx j$

[1] In first-order logic with equality, where equality between terms of sort int is $\approx_i$ and equality between terms of sort list is $\approx_l$.

(g) $p(\neg\mathsf{car}(x), x)$

(h) $\neg p(\mathsf{car}(x), x) \vee p(j, \mathsf{cons}(j, x))$

(i) $\neg p(b, x) \vee p(b, \mathsf{cons}(b, x)) \vee b$

(j) $\forall i : \mathsf{int}, \forall x : \mathsf{list}\ (\mathsf{cons}(\mathsf{car}(x), \mathsf{cdr}(x)) \approx_l x)$

(k) $\exists i : \mathsf{int}, \forall y : \mathsf{list}\ (\mathsf{cons}(b, p(x, y)) \approx_l \mathsf{cdr}(y))$

Please submit your solution until Tuesday, November 22, 2022 at 12:00. Joint solutions prepared by up to three persons are allowed. Please do not forget to write your name on your solution.

Submission possibilities:

- Use the folder Homework 04 in OLAT