Technische Universität Wien
Fakultät für Informatik
Uwe Egly
Antonius Weinzierl

WS 2013
Übungsblatt Block 2
Matrikelnummer: XXXXXXXX

Exercises on Formal Methods in Computer Science

If you would like to receive feedback *in the exercise sessions*, you should submit your solutions to TUWEL no later than *November 11th 2013*. You will get feedback in electronic form if you upload you exercises no later than *November 29th 2013*.

This exercise sheet is divided into two parts: first algorithms and techniques, second proofs and properties. Note that the questions of Block 2 in the final exam are going to have a strong emphasis on understanding and proofs. All exercises are relevant for the final exam. We strongly recommend solving *at least* the following exercises until the presentation of the solutions: Exercise 2 (a), Exercise 3 (a), (b), Exercise 6, Exercise 7 (b), Exercise 8 (a), (c), and Exercise 9 (a).

1 Algorithms and Techniques

Exercise 1 First-Order Theories

To get an intuition, what a formula means, it often helps to visualize an example instantiation of the occuring relations. That is, one visualizes a model (or interpretation) of the formula by drawing the respective relations. Binary relations can be visualized very easily as directed graphs: let $R \subseteq U \times U$ be a relation on the universe (domain) U, then the corresponding directed graph G is $G = (U, R)$. So, whenever two elements u_1 and u_2 of the universe of an interpretation are related by R, then the corresponding graph contains an edge between u_1 and u_2.
Consider the formula $\forall x \forall y \forall z : xRy \wedge yRz \rightarrow xRz$ and an interpretation I on the universe $U = \{u, v, w, t\}$ such that $I(R) = \{(u, v), (v, w), (u, w), (v, t), (u, t)\}$. Now $I(R)$ can be seen as a directed graph over U and this graph is shown in Figure 1. Since I is a model of the above formula, the shown graph is a visualization of this model.

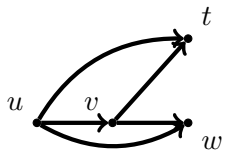


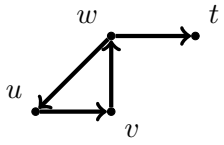
Figure 1: A graph visualizing a model of the formula $\forall x \forall y \forall z : xRy \wedge yRz \rightarrow xRz$.

(a) Let T_1 be a theory consisting of the following fomulae:

$$\forall x : xRx$$
$$\forall x \forall y : xRy \rightarrow yRx$$

i) Pick a domain of size at least 5, pick any model of T_1 based on your chosen domain, and visualize R.

ii) Consider the following graph, and extend it such that it corresponds to a model of T_1.



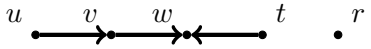
iii) Visualize a relation, which violates T_1.

(b) Visualize the theory T_2, which consists of the formula:

$$\forall x \exists y : xRy$$

i) Pick a domain of size at least 5 and visualize a chosen model of T_2.

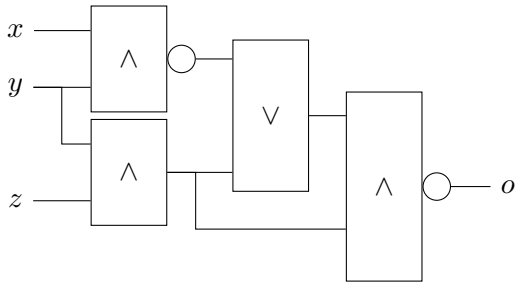
ii) Consider the following graph, and extend it such that it corresponds to a model of T_2.



Exercise 2 Tseitin Transformation

(a) For the formula $\psi = \big(a \rightarrow (b \rightarrow \neg a)\big)$ use Tseitin translation to compute a sat-equivalent CNF.

(b) Given the circuit below with AND, NAND, and OR gates, use Tseitin translation to obtain a linear-size (and sat-equivalent) CNF.



Exercise 3 Implication Graphs

Let $\mathcal{C}$ be a clause set consisting of the following clauses:

$$
\begin{aligned}
c_1 &: \quad (\neg A \vee B) \\
c_2 &: \quad (\neg A \vee \neg B \vee C) \\
c_3 &: \quad (A \vee B) \\
c_4 &: \quad (\neg F \vee \neg B \vee \neg G) \\
c_5 &: \quad (G \vee \neg E) \\
c_6 &: \quad (G \vee D) \\
c_7 &: \quad (C \vee E \vee \neg D) \\
c_8 &: \quad (\neg A \vee C)
\end{aligned}
$$

(a) Draw an implication graph for $\mathcal{C}$. Use the decision $C = 0@1$, and $F = 1@2$ until you reach a conflict.

(b) Determine all UIPs in the implication graph, find the first UIP and use resolution to learn a conflict clause corresponding to the first UIP.

(c) Add the learned clause, apply conflict-driven backtracking and draw the resulting implication graph.

Exercise 4 Sparse Method

Apply the Sparse Method including preprocessing on the formula φ below to obtain a propositional formula. Note that φ is not yet in NNF (Negation Normal Form).

$$(x_1 = x_2 \rightarrow x_2 = x_3) \wedge \left[\neg(x_2 = x_4 \vee x_3 \neq x_4 \vee x_4 \neq x_5) \vee (x_6 \neq x_5 \wedge x_6 = x_7 \wedge x_7 = x_3)\right]$$

Exercise 5 Ackermann's Reduction

Apply Ackermann's reduction on the following EUF-formula φ to obtain an E-formula:

$$F(F(x_1)) \neq F(x_1) \wedge G(x_1, x_2) = F(x_2) \wedge F(G(x_2, F(x_2))) \neq F(F(x_1))$$

2 Proofs and Properties

Exercise 6 First-Order Theories

In the lecture, we discussed reasoning under different theories. Here we are concerned with LISP-like lists and the theory $\mathcal{T}^E_{cons} = \mathcal{T}_{cons} \cup \mathcal{T}_E$. In a verification attempt of some program, we have to prove the following:

> *For non-atomic lists ℓ_1, ℓ_2, if the "car" of both lists are equal and the "cdr" of both lists are equal, then ℓ_1 is equal to ℓ_2.*

We formalize the above statement as follows:

$$\varphi\colon \quad \big[\, \neg atom(\ell_1) \wedge \neg atom(\ell_2) \wedge car(\ell_1) \doteq car(\ell_2) \wedge cdr(\ell_1) \doteq cdr(\ell_2) \,\big] \;\rightarrow\; \ell_1 \doteq \ell_2$$

Prove the statement $\mathcal{T}^E_{cons}$-valid, i.e., show that $\mathcal{T}^E_{cons} \models \varphi$.

Hint: Besides the equality axioms reflexivity, symmetry and transitivity, the following axioms from $\mathcal{T}^E_{cons}$ are sufficient for a proof:

(1) Substitution axioms (functional congruence) for *cons*:

$$\forall x_1 \forall x_2 \forall y_1 \forall y_2 \, [(x_1 \doteq x_2 \wedge y_1 \doteq y_2) \rightarrow cons(x_1, y_1) \doteq cons(x_2, y_2)]$$

(2) Construction:

$$\forall x \, [\neg atom(x) \rightarrow cons(car(x), cdr(x)) \doteq x]$$

Exercise 7 Tseitin Transformation

In the first part of this exercise, we consider a restriction of the Tseitin transformation where the input formula is only composed of propositional variables, negation, and conjunction. In the second part, we consider a simplified transformation whose output is not in CNF.

(a) Let ψ be a propositional formula and let $\delta(\psi)$ be the set of clauses resulting from Tseitin's transformation on ψ. Prove that the following holds:

If ψ is satisfiable then $\delta(\psi)$ is satisfiable.

You only need to prove this for the connectives $\wedge$ and $\neg$. Use the below clause schemes, which introduce a new label for every boolean variable.

$$\begin{array}{rrrr}
L_a \leftrightarrow a & (\neg L_a \vee a) & (L_a \vee \neg a) & \\
L_\phi \leftrightarrow (L_1 \wedge L_2) & (\neg L_\phi \vee L_1) & (\neg L_\phi \vee L_2) & (L_\phi \vee \neg L_1 \vee \neg L_2) \\
L_\phi \leftrightarrow \neg L_1 & (\neg L_\phi \vee \neg L_1) & (L_\phi \vee L_1) &
\end{array}$$

(b) Consider a simplified variant of Tseitin's transformation: let φ be a propositional formula, let $\Sigma(\varphi)$ be the set of all subformulas of φ, and let ℓ_φ be the label for φ. Then, the result of simplified Tseitin's transformation is the formula:

$$\lambda = \left(\bigwedge_{\psi \in \Sigma(\varphi)} (\ell_\psi \leftrightarrow \psi) \right) \rightarrow \ell_\varphi$$

Prove: λ is valid if and only if φ is valid.

Exercise 8 Implication Graphs

(a) Show that in a conflict graph the first UIP is uniquely defined, i.e., there is exactly one node in the implication graph which is a first UIP.

(b) Let $\mathcal{C}$ be a set of clauses and G a conflict graph with respect to $\mathcal{C}$. Prove: if C_l is the first clause that is learned following the first-UIP scheme, then C_l is a consequence of $\mathcal{C}$.

Bonus questions: how can this statement be used to show that all clauses that are learned (following the first-UIP scheme) are a consequence of $\mathcal{C}$?

(c) Prove: *During the run of a SAT solver, the implication graph G_k at step k is acyclic.*

Hints:

1) Perform a proof by induction over k.
2) Consider the following events that can occur:
   (i) making a decision,
   (ii) unit propagation (one step of BCP),
   (iii) a clause is unsatisfiable,
   (iv) backtracking.

Exercise 9 Ackermann's Reduction

(a) The removal of Boolean variables from an E-formula is defined as follows:

Definition. Let φ^E be any E-formula with Boolean variables $b_1, \ldots, b_n$. Construct an E-formula ψ^E without any Boolean variable by replacing each b_i by $v_{b_{i,1}} \doteq v_{b_{i,2}}$ where $v_{b_{i,1}}$, $v_{b_{i,2}}$ are two new term variables (identifiers).

Prove that φ^E is E-satisfiable iff ψ^E is E-satisfiable.

(b) Transform the EUF-formula φ^{EUF} below to an E-formula φ^E using Ackermann's reduction. Note that φ^{EUF} contains an uninterpreted predicate, which requires special treatment first.

$$\varphi^{EUF}: \quad F(F(x_1)) \doteq G(x_2, G(x_1, x_3, x_4), F(x_2)) \rightarrow p(x_1, y).$$