Software Engineering and Programming Languages
Prof. Dr. Michael Leuschel
David Geleßus, Fabian Vu

Introduction to Logic Programming – WS 2023
Solutions for Exercise Sheet 7

1 Exercises

Exercise 2 (CNF in FOL)
Transform the following predicate to conjunctive normalform.

$\forall n : (n \in \mathbb{N} \Rightarrow (\exists m : m \in \mathbb{N} \land m > n))$

$\equiv \forall n : (n \notin \mathbb{N} \lor (\exists m : m \in \mathbb{N} \land m > n))$ introduce skolem function $f(n) := m$

$\equiv \forall n : (n \notin \mathbb{N} \lor (f(n) \in \mathbb{N} \land f(n) > n)$

$\equiv \forall n : ((n \notin \mathbb{N} \lor f(n) \in \mathbb{N}) \land (n \notin \mathbb{N} \lor f(n) > n))$

Set representation: $\{\{n \notin \mathbb{N}, f(n) \in \mathbb{N}\}, \{n \notin \mathbb{N}, f(n) > n\}\}$

We have to introduce a skolem function since the existentially quantified variable m is used in a predicate together with the universally quantified variable n.

Exercise 3 (Proof by Contradiction, CNF in FOL)
Prove by contradiction that the following predicate is true:

$$(\forall x : (\neg P(x) \Rightarrow Q(x)) \wedge \exists x : \neg Q(x)) \Rightarrow \exists x : P(x) \tag{1}$$

Note that there are three different variables called `x` which are locally quantified.

$\neg[(\forall x : (\neg P(x) \Rightarrow Q(x)) \wedge \exists x : \neg Q(x)) \Rightarrow \exists x : P(x)]$

$\equiv \neg[(\neg(\forall x : (\neg P(x) \Rightarrow Q(x)) \wedge \exists x : \neg Q(x))) \vee \exists x : P(x)]$

$\equiv \neg[(\neg(\forall x : (P(x) \vee Q(x)) \wedge \exists x : \neg Q(x))) \vee \exists x : P(x)]$

$\equiv (\forall x : (P(x) \vee Q(x)) \wedge \exists x : \neg Q(x)) \wedge \neg\exists x : P(x)$

$\equiv (\forall x : (P(x) \vee Q(x)) \wedge \exists x : \neg Q(x)) \wedge \forall x : \neg P(x)$ introduce skolem constant $s := x$

$\equiv (\forall x : (P(x) \vee Q(x)) \wedge \neg Q(s)) \wedge \forall x : \neg P(x)$ rename universally quantified variables

$\equiv (\forall x : (P(x) \vee Q(x)) \wedge \neg Q(s)) \wedge \forall y : \neg P(y)$ move universal quantifiers to the left

$\equiv \forall x, y : (P(x) \vee Q(x)) \wedge \neg Q(s) \wedge \neg P(y)$

The formula has to be true for each x and y. Choose $x = y = s$.

$(P(s) \vee Q(s)) \wedge \neg Q(s) \wedge \neg P(s) \equiv \perp$

This formula is a contradiction and the assignment $x = y = s$ is a counterexample. We have thus proven the formula by contradiction.

Note that the skolemization preserves unsatisfiability of formulae. We thus have to negate the formula before skolemization.

Exercise 4 (Proof by Contradiction, CNF and Resolution in FOL)
Define the following statements in predicate logic:

- For all humans it applies: If they lie in the sun for too long and don't use sunscreen, they get sunburn.
- All children are humans.
- Billy is a child.
- Billy has been in the sun too long and doesn't use sunscreen.

Note: You can use abbreviations for the predicate names.

Prove by contradiction and resolution that Billy gets sunburn. Specify the most general unificator for each resolution step.

1. $\forall x : human(x) \wedge sun\_too\_long(x) \wedge \neg used\_sunscreen(x) \Rightarrow sunburn(x)$
   $\equiv \forall x : \neg human(x) \vee \neg sun\_too\_long(x) \vee used\_sunscreen(x) \vee sunburn(x)$
2. $\forall x : child(x) \Rightarrow human(x) \equiv \forall x : \neg child(x) \vee human(x)$
3. $child(billy)$
4. $sun\_too\_long(billy)$.
5. $\neg used\_sunscreen(billy)$.
6. $\neg sunburn(billy)$. (negated goal)
7. $(1.) \wedge (6.) \xrightarrow{resolution} \neg \mathbf{human}(\mathbf{billy}) \vee \neg \mathbf{sun\_too\_long}(\mathbf{billy}) \vee \mathbf{used\_sunscreen}(\mathbf{billy})$
   mgu: $\{X/billy\}$
8. $(5.) \wedge (7.) \xrightarrow{resolution} \neg \mathbf{human}(\mathbf{billy}) \vee \neg \mathbf{sun\_too\_long}(\mathbf{billy})$
   mgu: $\{\}$
9. $(4.) \wedge (8.) \xrightarrow{resolution} \neg \mathbf{human}(\mathbf{billy})$
   mgu: $\{\}$
10. $(2.) \wedge (9.) \xrightarrow{resolution} \neg \mathbf{child}(\mathbf{billy})$
    mgu: $\{X/billy\}$
11. $(3.) \wedge (10.) \equiv \bot$
    mgu: $\{\}$

This is a linear resolution since we always use the last formula obtained by resolution for the next step.