Logical reasoning and programming, lab session 7

(November 6, 2023)

Instead of installing all the theorem provers on your computer, you may experiment with them using System on TPTP.

7.1 Unify the following pairs of formulae:

(a) $\{p(X,Y) \doteq p(Y,f(Z))\}$,
(b) $\{p(a,Y,f(Y)) \doteq p(Z,Z,U)\}$,
(c) $\{p(X,g(X)) \doteq p(Y,Y)\}$,
(d) $\{p(X,g(X),Y) \doteq p(Z,U,g(U))\}$,
(e) $\{p(g(X),Y) \doteq p(Y,Y), p(Y,Y) \doteq p(U,f(W))\}$.

Note: You can check your results in SWISH using `unify_with_occurs_check/2`.

7.2 What is the size of the maximal term that is produced when you try to unify

$$\{f(g(X_1,X_1),g(X_2,X_2),\ldots,g(X_{n-1},X_{n-1})) \doteq f(X_2,X_3,\ldots,X_n\}.$$

7.3 We say that a binary predicate q is the transitive closure of a binary predicate p, if $q(s,t)$ iff there is a sequence of terms $s=t_1,t_2,\ldots,t_{n-1},t_n=t$ such that $p(t_i,t_{i+1})$, for $1 \le i < n$. Is the formula

$$\forall X \forall Z (q(X,Z) \leftrightarrow (p(X,Z) \vee \exists Y (p(X,Y) \wedge q(Y,Z))))$$

a correct definition of q?

7.4 The compactness theorem in First-Order Logic says that a set of sentences has a model iff every finite subset of it has a model. Use this theorem to show that the transitive closure is not definable in FOL.

Hint: Assume for a contradiction that φ is a formula that expresses that q is the transitive closure of p. Let $\psi^n(a,b) = \neg(\exists X_1 \ldots \exists X_{n-1}(p(a,X_1) \wedge p(X_1,X_2) \wedge \cdots \wedge p(X_{n-1},b))$ (Hence $\psi^1(a,b)$ means $\neg p(a,b)$ and $\psi^2(a,b)$ means $\neg(\exists X_1(p(a,X_1) \wedge p(X_1,b))))$. What can you say about the satisfiability of $\Gamma = \{\varphi\} \cup \{q(a,b)\} \cup \{\psi^1(a,b),\psi^2(a,b),\ldots\}$?

7.5 Show that the resolution rule is correct.

7.6 Derive the empty clause $\Box$ using the resolution calculus from:

(a) $\{\{\neg p(X), \neg p(f(X))\}, \{p(f(X)), p(X)\}, \{\neg p(X), p(f(X))\}\}$
(b) $\{\{\neg p(X,a), \neg p(X,Y), \neg p(Y,X)\}, \{p(X,f(X)), p(X,a)\}, \{p(f(X),X), p(X,a)\}\}$

7.7 Prove using the resolution calculus that from

$$\forall X \forall Y (p(X,Y) \rightarrow p(Y,X))$$
$$\forall X \forall Y \forall Z ((p(X,Y) \wedge p(Y,Z)) \rightarrow p(X,Z))$$
$$\forall X \exists Y \, p(X,Y)$$

follows $\forall X p(X,X)$.

7.8 Check PyRes; simple resolution-based theorem provers for first-order logic. You can find proofs for the previous examples using them. For example, use

```
pyres-fof.py -tifb -HPickGiven5 -nlargest
```

There are various heuristics (`FIFO`, `SymbolCount`, `PickGiven5`, and `PickGiven2`) and literal selections (`first`, `smallest`, `largest`, `leastvars`, and `eqleastvars`) available. Use `-p` to see a proof.

7.9 List all the possible applications of the factoring rule on the clause

$$\{p(X, f(Y), Z), p(T, T, g(a)), p(f(b), S, g(W)), \neg s(Z, T), \neg s(c, d)\}.$$

If it is possible to use the factoring rule several times, then produce even these results.

7.10 Produce all the possible paramodulants, but do not perform paramodulations into variables, of

$$\{\{p(X), \neg q(X, Y), f(c, Y) = g(X)\}, \{p(Z), q(g(a), f(Z, b)), c = f(c, c)\}\}.$$

7.11 Formulate the following problems in the TPTP language and (dis)prove them using the E prover. Assuming the following group axioms

$$\begin{aligned} &e \cdot X = X, \\ &X^{-1} \cdot X = e, \\ &(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z) \end{aligned}$$

your task is to (dis)prove

(a) $X \cdot e = X$,

(b) $X \cdot X^{-1} = e$,

(c) $X \cdot Y = Y \cdot X$,

(d) $X \cdot Y = Y^{-1} \cdot X^{-1}$.

7.12 Use the model finder Paradox to produce counterexamples for unprovable claims in the previous exercise **7.11**.

7.13 Use PyRes to prove **7.11**a. Note that PyRes uses the naïve handling of equality. For example, use

```
pyres-fof.py -tifb -HPickGiven5 -nlargest
```

There are various heuristics (`FIFO`, `SymbolCount`, `PickGiven5`, and `PickGiven2`) and literal selections (`first`, `smallest`, `largest`, `leastvars`, and `eqleastvars`) available. Use `-p` to see a proof.

7.14 Formalize in the TPTP format a simple example with the following axioms

$$\forall X \neg r(X, X),$$
$$\forall X \forall Y \forall Z (r(X, Y) \land r(Y, Z) \to r(X, Z)),$$
$$\forall X \exists Y r(X, Y)$$

and check how fast can Paradox generate possible finite models for this simple problem. Clearly, it will never find a model, because the problem has only infinite models.

7.15 Try the Vampire prover on the problem GRP140-1 from the TPTP library. We demonstrate the effect of the limited resource strategy (LRS), which discards unprocessed clauses that will be unlikely processed in a given time limit, by this example. For the intended behavior you need a special setting—age:weight ratio is 5:1 and the forward subsumption is turned off:

```
vampire -awr 5:1 -fsr off -t 30 GRP140-1.p
```

First, try the timelimit 30s, then try 15s, 7s, ... You can try even shorter times than 1s, e.g., `-t 5d` means 5 deciseconds.

For comparison you can try the competition mode on the same problem

```
vampire --mode casc GRP140-1.p
```

7.16 Try the E prover on the problem GRP001-1 from the TPTP library. Compare how can the use of a literal selection strategy influence the behavior of the prover:

```
eprover --literal-selection-strategy=NoSelection GRP001-1.p
eprover --literal-selection-strategy=SelectLargestNegLit \
          GRP001-1.p
```

You may also visualize the proof using the Interactive Derivation Viewer (IDV) tool for graphical rendering of derivations through System on TPTP.