Intro to TPTP

B4B36ZUI

April 6, 2020

1 Introduction

The goal of this document is to (very) briefly introduce the TPTP syntax for the purposes of the B4B36ZUI course (and not to completely explain everything as it would be really out of the scope of this document).

2 Example with comments

This introduction is taken (with few changes) from http://www.tptp.org/Seminars/TPTPWorldTutorial/LogicFOF.html.

2.1 Main elements

1. predicates
2. constants
3. functions
4. universally quantified variables with ∀ (for all)
5. existentially quantified variables with ∃ (there exists)

2.2 Problem

- Axioms
  - The USA is a country.
  - The capital of every country is a beautiful big city
  - There is some city that is the capital of the USA
  - Every big city has crime.
  - A big city is a city.
- Conjecture
  - Therefore, there is some city that is beautiful but has crime.

2.3 Logic

This can be translated to first order logic as:

- Axioms
  - $\text{country}(usa)$
  - $\forall C(\text{country}(C) \implies (\text{big\_city}(\text{capital\_of}(C)) \wedge \text{beautiful}(\text{capital\_of}(C))))$
  - $\exists C(\text{city}(C) \wedge C = \text{capital\_of}(USA))$
  - $\forall C(\text{big\_city}(C) \implies \text{has\_crime}(C))$
  - $\forall(\text{big\_city}(C) \implies \text{city}(C))$
- Conjecture
  - $\exists C(\text{city}(C) \wedge \text{beatiful}(C) \wedge \text{has\_crime}(C))$

2.4 TPTP

The logic can be translated to the TPTP language as:

```
1  fof(usa,axiom, country(usa) ).
2  fof(country_big_city,axiom,
3      ! [C] : ( country(C) => ( big_city(capital_of(C)) & beautiful(capital_of(C)) ) ) ).
4  fof(usa_capital_axiom,axiom,
5      ? [C] : ( city(C) & C = capital_of(usa) ) ).
6  fof(crime_axiom,axiom,
7      ! [C] : ( big_city(C) => has_crime(C) ) ).
8  fof(big_city_city,axiom,
9      ! [C] : ( big_city(C) => city(C) ) ).
10
11 fof(some_beautiful_crime,conjecture,
12     ? [C] : ( city(C) & beautiful(C) & has_crime(C) ) ).
```

2.4.1 Description of main parts

- `fof(usa,axiom, country(usa) ).`
  - `fof` determines the type of the formula and stands for *first order logic formula*, there are other types available — e.g., *typed first-order logic formulas* (`tff`), *typed higher-order logic formulas* (`thf`) — but those are out of the scope of this course (if interested, short examples are available at http://www.tptp.org/Seminars/TPTPWorldTutorial/LogicTFF.html and http://www.tptp.org/Seminars/TPTPWorldTutorial/LogicTHF.html).
  - `usa` is an arbitrary name of the formula (just keep it unique)
  - `axiom` stands for the *formula type* — this describes the facts. We are going to use only another type and that is `conjecture` which will describe what we want to prove.
- `fof(big_city_city,axiom, ! [C] : ( big_city(C) => city(C) ) ).`
  - `!` is the *universal quantifier* $\forall$ (the *existential* $\exists$ is denoted using `?`)
  - `C` is a *variable* (variables are denoted using **capital** letters)
  - `big_city(C)` is a *function* with variable `C`
  - `=>` is the *implication* operator $\Longrightarrow$
- `fof(some_beautiful_crime,conjecture, ? [C] : ( city(C) & beautiful(C) & has_crime(C) ) ).`
  - `conjecture` stands for the *formula type* — here it denotes that this is the formula we want to prove
  - `?` is the *existential quantifier* $\exists$
  - `&` is the *logical and* operator $\wedge$ (`|` represents *logical or* $\vee$, $\sim$ represents *negation* $\neg$, `<=>` represents *equivalence* $\Longleftrightarrow$, `=` represents *equality* operator, and `!=` *inequality* operator)

3 Running the prover

The easiest way to attempt finding the proof of the problem is through the SystemOnTPTP webpage where you can insert your problem description directly and run a prover.

It is recommended to start with the prover named Prover9 (this is the prover that is going to be used for checking your solutions to the assignment) and to generally use the time-limit settings responsibly as the resources are shared and limited.

4 Common errors/troubleshooting

The TPTP has a lot of parentheses, it is recommended to use an editor that highlights closing and opening parentheses as it is one of the more common errors. Another common error is incorrect understanding of priority of operators — it is recommended to use rather more of parentheses than less to avoid the priority problems.

Another common problem is usage of variables without quantifiers (∃, ∀) — always write the required quantifiers.

The provers are based on the Open-world assumption thus do not forget to state all the required facts in the code (e.g., that constants represent different individuals: `fof(diff_alice_bob, axiom, (alice != bob)).`).v.

5 See also

It is strongly recommended to go through the TPTP project webpage http://www.tptp.org/ (esp. the QuickGuide where more details are available. If unsure about the syntax, the most comprehensive (but not really user friendly) is definition available at http://www.tptp.org/TPTP/SyntaxBNF.html — it is, however, recommended to go through examples available at http://www.tptp.org/cgi-bin/SeeTPTP?Category=Problems, especially the PUZ and PLA domains. For general overview of the situational calculus, see https://artint.info/2e/html/ArtInt2e.Ch15.S1.SS1.html.