Logics Exercise

TU München
Institut für Informatik

Prof. Tobias Nipkow
Dr. Peter Lammich
Simon Wimmer

SS 2016 | Exercise Sheet 10 | 15.06.2016

Submission of Homework: Before tutorial on June 22

Exercise 10.1. [Proofs in Sequent Calculus]
Using sequent calculus, prove or disprove wether the following formulas are tautologies:

- $A \vee \neg A$
- $((P \to Q) \to P) \to Q$
- $\neg(A \wedge B) \to \neg A \vee \neg B$

Also give the corresponding tableau for the last formula.

Exercise 10.2. [Modified Calculi]
In which ways does the sequent calculus change if we make one of the following modifications?

- We restrict the axiom for formulas to atoms, i.e. $A, \Gamma \Rightarrow A, \Delta$.
- We replace the axioms by $F \Rightarrow F$ and $\bot \Rightarrow \emptyset$ and add the weakening rule $\dfrac{\Gamma \Rightarrow \Delta}{\Gamma, \Gamma' \Rightarrow \Delta, \Delta'}$ to the calculus.
- We replace $\vee_R$ by $\dfrac{\Gamma \Rightarrow A, \Delta}{\Gamma \Rightarrow A \vee B, \Delta}$ and $\dfrac{\Gamma \Rightarrow B, \Delta}{\Gamma \Rightarrow A \vee B, \Delta}$.

Exercise 10.3. [Derived Rule]
Show that if $\vdash_G \ \Gamma \Rightarrow \neg X, \Delta$ then $\vdash_G \ X, \Gamma \Rightarrow \Delta$

Homework 10.1. [Hintikka's Lemma] (5 points)
For this exercise, we assume the set of basic connectives is $\neg, \vee, \wedge$. A set of formulas H is called Hintikka-set, iff

1. For any atom A, not both $A \in H$ and $\neg A \in H$
2. If $\neg\neg Z \in H$ then also $Z \in H$
3. If $F_1 \wedge F_2 \in H$ then also $F_1 \in H$ and $F_2 \in H$
4. If $\neg(F_1 \vee F_2) \in H$ then also $\neg F_1 \in H$ and $\neg F_2 \in H$
5. If $F_1 \vee F_2 \in H$ then also $F_1 \in H$ or $F_2 \in H$
6. If $\neg(F_1 \wedge F_2) \in H$ then also $\neg F_1 \in H$ or $\neg F_2 \in H$

Show: Every Hintikka-set is satisfiable.

Homework 10.2. [Sequent-Calculus] (5 points)
Prove or disprove the following formulas in sequent calculus. For invalid formulas, read off a counterexample from the stuck proof tree:

1. $A \wedge (B \vee C) \longrightarrow (A \wedge B) \vee (A \wedge C)$
2. $\neg(A \wedge B) \longrightarrow \neg A \wedge \neg B$

Homework 10.3. [Sequent Prover] (10 points)
Implement a sequent calculus prover in your favorite programming language, and test it for all examples from this exercise sheet. Submission: Source code for prover and tests, README file containing instructions how to build prover and reproduce tests, as tgz-file by email to Simon or Peter.

Hint: You do not need to implement a parser, it's enough to specify the test-cases in a source-file. You also do not need to reconstruct counterexamples or proof-trees, a result valid/invalid is enough.