Theorie der Informatik

G. Röger
Frühjahrssemester 2020

Universität Basel
Fachbereich Informatik

Übungsblatt 2 — Lösungen

Aufgabe 2.1 (Äquivalenzen; 1.5+1.5 Punkte)

(a) Verwenden Sie die Äquivalenzen aus der Vorlesung um die folgende Formel in KNF zu bringen. Wenden Sie in jedem Schritt nur eine Äquivalenz an und geben Sie diese an.

$$\phi = (\neg(A \leftrightarrow \neg B) \rightarrow C)$$

Lösung:

$$\begin{aligned}
\phi &= (\neg(A \leftrightarrow \neg B) \rightarrow C) & \\
&\equiv (\neg\neg(A \leftrightarrow \neg B) \lor C) & (\rightarrow\text{-Eliminierung}) \\
&\equiv ((A \leftrightarrow \neg B) \lor C) & (\text{Doppelnegation}) \\
&\equiv (((A \rightarrow \neg B) \land (\neg B \rightarrow A)) \lor C) & (\leftrightarrow\text{-Eliminierung}) \\
&\equiv (((\neg A \lor \neg B) \land (\neg B \rightarrow A)) \lor C) & (\rightarrow\text{-Eliminierung}) \\
&\equiv (((\neg A \lor \neg B) \land (\neg\neg B \lor A)) \lor C) & (\rightarrow\text{-Eliminierung}) \\
&\equiv (((\neg A \lor \neg B) \land (B \lor A)) \lor C) & (\text{Doppelnegation}) \\
&\equiv (C \lor ((\neg A \lor \neg B) \land (B \lor A))) & (\text{Kommutativität}) \\
&\equiv ((C \lor (\neg A \lor \neg B)) \land (C \lor (B \lor A))) & (\text{Distributivität})
\end{aligned}$$

(b) Zeigen Sie, dass die folgende Formel eine Tautologie ist, indem Sie zeigen, dass $\phi \equiv (A \lor \neg A)$ gilt. Verwenden Sie die Äquivalenzen aus der Vorlesung, wenden Sie in jedem Schritt nur eine Äquivalenz an und geben Sie diese an.

$$\phi = (A \lor (\neg(A \land \neg(\neg A \land C)) \lor (A \land B)))$$

Lösung:

$$\begin{aligned}
\phi &= (A \lor (\neg(A \land \neg(\neg A \land C)) \lor (A \land B))) & \\
&\equiv (A \lor ((A \land B) \lor \neg(A \land \neg(\neg A \land C)))) & (\text{Kommutativität}) \\
&\equiv ((A \lor (A \land B)) \lor \neg(A \land \neg(\neg A \land C))) & (\text{Assoziativität}) \\
&\equiv (A \lor \neg(A \land \neg(\neg A \land C))) & (\text{Absorption}) \\
&\equiv (A \lor (\neg A \lor \neg\neg(\neg A \land C))) & (\text{De Morgan}) \\
&\equiv (A \lor (\neg A \lor (\neg A \land C))) & (\text{Doppelnegation}) \\
&\equiv (A \lor \neg A) & (\text{Absorption})
\end{aligned}$$

Aufgabe 2.2 (Inferenz; 1+1+1 Punkte + 1 Bonuspunkt)

Auf der Vorlesungsseite finden Sie ein Javaprogramm zum Überprüfen aussagenlogischer Beweise. Verwenden Sie das Programm, um die folgenden Aussagen zu beweisen. Für eine Aussage der Form WB $\models \varphi$, schreiben Sie dazu eine Ableitung in eine Textdatei, die nur Formeln aus WB als Annahmen verwendet und deren letzte Zeile φ ist. Ein Beispiel dafür finden Sie in der Datei `proof.txt`.

Das Programm überprüft, dass WB $\vdash \varphi$ gilt. Da der im Program verwendete Kalkül korrekt ist, folgt daraus auch WB $\models \varphi$.

(a) $\{A, B\} \models ((A \wedge B) \vee C)$

Lösung:

```
A                    | Assumption    |
B                    | Assumption    |
(A /\ B)             | AndIntro      | 1, 2
((A /\ B) \/ C)      | OrIntroLeft   | 3
```

(b) $\{(A \wedge B)\} \models (A \to (B \vee C))$

Lösung:

```
(A /\ B)             | Assumption        |
B                    | AndElimLeft       | 1
(B \/ C)             | OrIntroLeft       | 2
(~A \/ (B \/ C))     | OrIntroRight      | 3
(A -> (B \/ C))      | ImplicationIntro  | 4
```

(c) $\{((A \vee B) \to (A \to C)), A\} \models C$

Lösung:

```
A                          | Assumption   |
(A \/ B)                   | OrIntroLeft  | 1
((A \/ B) -> (A -> C))     | Assumption   |
(A -> C)                   | ModusPonens  | 2, 3
C                          | ModusPonens  | 1, 4
```

(d) $\{((C \vee D) \leftrightarrow (A \wedge B)), \neg E, (((A \wedge B) \wedge (C \vee D)) \to E)\} \models \neg(A \wedge B)$
Ergänzen Sie dazu den Kalkül um eine neue Regel *Negations-Einführung*:

$$\frac{(\varphi \to \psi), (\varphi \to \neg\psi)}{\neg\varphi}$$

Lösung:

```
((C \/ D) <-> (A /\ B))          | Assumption                |
((A /\ B) -> (C \/ D))           | BiimplicationElimLeft     | 1
~E                               | Assumption                |
(((A /\ B) /\ (C \/ D)) -> E)    | Assumption                |
~((A /\ B) /\ (C \/ D))          | ModusTollens              | 3, 4
(~(A /\ B) \/ ~(C \/ D))         | DeMorgan2LeftToRight      | 5
((A /\ B) -> ~(C \/ D))          | ImplicationIntro          | 6
~(A /\ B)                        | NegationIntro             | 2, 7
```

Dazu muss die Regel mit folgender Zeile in `Calculus.java` eingefügt werden:

```
addRule("NegationIntro", "(X -> Y), (X -> ~Y) |- ~X");
```

(e) *Bonusaufgabe:* Um die Korrektheit des Kalküls zu beweisen muss die Korrektheit aller Regeln gezeigt werden. Zeigen Sie die Korrektheit der Regel *Negations-Einführung*.

Lösung:

Sei $\mathcal{I}$ ein Modell von $(\varphi \rightarrow \psi)$ und $(\varphi \rightarrow \neg\psi)$. Aus $\mathcal{I} \models (\varphi \rightarrow \psi)$ folgt, dass $\mathcal{I} \not\models \varphi$ oder $\mathcal{I} \models \psi$ gelten muss. Angenommen $\mathcal{I} \models \varphi$, dann muss $\mathcal{I} \models \psi$ gelten (*). Aus $\mathcal{I} \models (\varphi \rightarrow \neg\psi)$ folgt aber, dass $\mathcal{I} \not\models \varphi$ oder $\mathcal{I} \models \neg\psi$ gelten muss. Da $\mathcal{I} \models \varphi$ laut unserer Annahme gilt, muss $\mathcal{I} \models \neg\psi$ gelten. Das ist ein Widerspruch zu (*). Die Annahme ($\mathcal{I} \models \varphi$) ist also falsch und es gilt $\mathcal{I} \models \neg\varphi$.

Da für alle Modelle $\mathcal{I}$ von $(\varphi \rightarrow \psi)$ und $(\varphi \rightarrow \neg\psi)$ auch $\mathcal{I} \models \neg\varphi$ gilt, folgt die Korrektheit der Regel: $\{(\varphi \rightarrow \psi), (\varphi \rightarrow \neg\psi)\} \models \neg\varphi$.

Hinweis zur Abgabe: Geben Sie bitte pro Teilaufgabe eine Textdatei ab, welche die Ableitung enthält. Die Datei muss vom Programm lesbar sein und als korrekte Ableitung der Aussage erkannt werden. Die neu eingefügte Regel (*Negations-Einführung*) benötigt eine neue Zeile im Programm. Kopieren Sie diese Zeile auf Ihre reguläre Abgabe. Die Bonusaufgabe kann nicht mit dem Programm gelöst werden.

Aufgabe 2.3 (Widerlegungstheorem; 2 Punkte)

Beweisen Sie das Widerlegungstheorem. Zeigen Sie also, dass für beliebige Formelmengen WB und Formeln φ folgendes gilt:

$$\text{WB} \cup \{\varphi\} \text{ ist unerfüllbar gdw. } \text{WB} \models \neg\varphi.$$

Lösung:

"$\Rightarrow$": Wenn $\text{WB} \cup \{\varphi\}$ unerfüllbar ist, dann gibt es keine Interpretation $\mathcal{I}$ mit $\mathcal{I} \models \text{WB}$ and $\mathcal{I} \models \varphi$. Daher gilt für all $\mathcal{I}$ mit $\mathcal{I} \models \text{WB}$, dass $\mathcal{I} \not\models \varphi$, und wir schliessen, dass $\text{WB} \models \neg\varphi$.

"$\Leftarrow$": Wenn $\text{WB} \models \neg\varphi$, dann gilt für alle $\mathcal{I}$ mit $\mathcal{I} \models \text{WB}$, dass $\mathcal{I} \models \neg\varphi$ und damit, dass $\mathcal{I} \not\models \varphi$. Daher gibt es keine Interpretation $\mathcal{I}$ mit $\mathcal{I} \models \text{WB} \cup \{\varphi\}$, also ist $\text{WB} \cup \{\varphi\}$ unerfüllbar.

Aufgabe 2.4 (Widerlegungsvollständigkeit; 2 Punkte)

Sei P ein Computerprogramm, das als Eingabe eine Menge von aussagenlogischen Formeln nimmt, und ausgibt, ob die eingegebene Formelmenge unerfüllbar ist.
Wie können Sie P verwenden, um für eine Wissensbasis WB und eine aussagenlogische Formel φ zu entscheiden, ob

(a) WB erfüllbar ist?

Lösung:

Wir lassen das Programm mit Eingabe WB laufen. Dann ist WB erfüllbar gdw. das Programm ausgibt, dass die Eingabe nicht unerfüllbar ist.

(b) $\text{WB} \models \varphi$?

Lösung:

Wir lassen das Programm mit Eingabe $\text{WB} \cup \{\neg\varphi\}$ laufen. Wegen des Widerlegungstheorems gilt $\text{WB} \models \varphi$ gdw. das Programm ausgibt, dass die Eingabe unerfüllbar ist.

(c) WB eine Tautologie ist?

Lösung:

Eine Wissenbasis WB ist eine Tautologie, genau dann wenn jede Interpretation ein Modell aller Formeln $\varphi \in \mathrm{WB}$ ist, und damit auch ein Modell der Konjunktion $(\bigwedge_{\varphi \in \mathrm{WB}} \varphi)$. WB ist also eine Tautologie gdw. es keine Interpretation $\mathcal{I}$ gibt, so dass $\mathcal{I} \not\models (\bigwedge_{\varphi \in \mathrm{WB}} \varphi)$. Das heisst, WB ist eine Tautologie gdw. $\neg(\bigwedge_{\varphi \in \mathrm{WB}} \varphi)$ unerfüllbar ist. Damit ist WB eine Tautologie, gdw. das Programm auf Eingabe $\{\neg(\bigwedge_{\varphi \in \mathrm{WB}} \varphi)\}$ ausgibt, dass die Formel nicht unerfüllbar ist.

(d) WB falsifizierbar ist?

Lösung:

Eine Wissensbasis ist falsifizierbar gdw. sie keine Tautologie ist. Ob sie eine Tautologie ist, können wir wie in der vorherigen Teilaufgabe entscheiden.