

Übungszettel 14 - Bonuszettel

Abgabetermin: 11. Februar 2023, 18:00 Uhr. Die Abgabe erfolgt über Ihr Ilias-Tutorium.

Aufgabe 1. Der Sequenzenkalkül mit den einfachen Regeln $\forall\text{-L}$ und $\exists\text{-R}$ ist nicht vollständig. Um die Vollständigkeit zu erreichen, kann man $\forall\text{-L}$ und $\exists\text{-R}$ durch die ihre copy-Varianten (die quantifizierte Formel wird vor der Instanziierung kopiert) ersetzen.

Zeigen Sie, dass man auch ohne diese copy-Varianten auskommen könnte, wenn man alternativ

- die Regel contract-L,
- oder die cut-Regel

benutzen darf. Die Behauptung ist also, dass der Sequenzenkalkül mit den einfachen $\forall\text{-L}$ und $\exists\text{-R}$ -Regeln vollständig wird, wenn man *contract-L* oder *cut* benutzen darf.

(3+3 Punkte)

Aufgabe 2. In der Vorlesung wurde gezeigt, wie man ein Dominoproblem in eine Menge von Axiomen A_1, A_2, A_3 übersetzen kann, so dass das Dominoproblem genau dann lösbar ist, wenn aus A_1 und A_2 die Aussage A_3 folgt. Betrachten Sie nun das konkrete Dominoproblem $D = \{\langle 1, 01 \rangle, \langle 0, 01 \rangle, \langle 010, \varepsilon \rangle\}$.

- (a) Geben Sie für dieses Problem eine Lösung an (1 Pkt)
(b) Modellieren Sie das Problem durch eine Menge von prädikatenlogische Formeln, die genau dann widersprüchlich sind, wenn D eine Lösung hat. (2 Pkt)
(c) Demonstrieren Sie die Lösbarkeit durch einen Resolutionsbeweis. (3 Pkt)

Aufgabe 3. Graphen modellieren wir durch eine zweistellige Relation $E(x, y)$, die ausdrücken soll, dass es eine Kante (engl.: *edge*) von x zu y gibt. Falls $E(x, y)$, sagt man auch, dass y ein Nachbar von x ist. Zeigen Sie, dass die Eigenschaft „*Jeder Knoten hat nur endlich viele Nachbarn*“ nicht in der Prädikatenlogik ausdrückbar ist.

(4 Punkte)

Für jedes n gibt es eine Formel $F_n(x)$
„Knoten x hat mind. n Nachbarn“

ZB x sei Knoten

$F_3(x)$ sei δ die Anzahl der Nachbarn

$$\begin{aligned} F_3(x) = & \exists x_1 \exists x_2 \exists x_3 (E(x, x_1) \wedge E(x, x_2) \wedge E(x, x_3) \\ & \wedge \neg(x=x_1) \wedge \neg(x=x_2) \wedge \neg(x=x_3)) \end{aligned}$$

$$\Rightarrow F_n(x) = \dots$$

Aufgabe 1. Der Sequenzenkalkül mit den einfachen Regeln $\forall\text{-L}$ und $\exists\text{-R}$ ist nicht vollständig. Um die Vollständigkeit zu erreichen, kann man $\forall\text{-L}$ und $\exists\text{-R}$ durch die ihre copy-Varianten (die quantifizierte Formel wird vor der Instanziierung kopiert) ersetzen.

Zeigen Sie, dass man auch ohne diese copy-Varianten auskommen könnte, wenn man alternativ

- die Regel contract-L,
- oder die cut-Regel

benutzen darf. Die Behauptung ist also, dass der Sequenzenkalkül mit den einfachen $\forall\text{-L}$ und $\exists\text{-R}$ -Regeln vollständig wird, wenn man *contract-L* oder *cut* benutzen darf.

$$\text{contract L: } \frac{\Delta, \forall x. F, F[t/x] \vdash \Gamma}{\Delta, \forall x. F \vdash \Gamma}$$

contract

$$\frac{\Delta \vdash \exists x. F, F[t/x], \Gamma}{\Delta \vdash \exists x. F, \Gamma}$$

contract - R aus contract - L

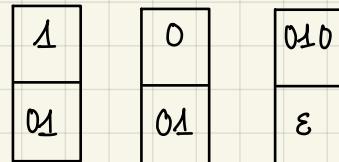
$$\frac{\begin{array}{c} \Delta \vdash G, G, \Gamma \\ \Delta, \neg G \vdash G, \Gamma \\ \Delta, \neg G, \neg G \vdash \Gamma \\ \Delta, \neg G \vdash \Gamma \\ \Delta \vdash G, \Gamma \end{array}}{\Delta \vdash G, \Gamma}$$

Aufgabe 2. In der Vorlesung wurde gezeigt, wie man ein Dominoproblem in eine Menge von Axiomen A_1, A_2, A_3 übersetzen kann, so dass das Dominoproblem genau dann lösbar ist, wenn aus A_1 und A_2 die Aussage A_3 folgt. Betrachten Sie nun das konkrete Dominoproblem $D = \{\langle 1, 01 \rangle, \langle 0, 01 \rangle, \langle 010, \varepsilon \rangle\}$.

- Geben Sie für dieses Problem eine Lösung an (1 Pkt)
- Modellieren Sie das Problem durch eine Menge von prädikatenlogische Formeln, die genau dann widersprüchlich sind, wenn D eine Lösung hat. (2 Pkt)
- Demonstrieren Sie die Lösbarkeit durch einen Resolutionsbeweis. (3 Pkt)

a, Lösung für das Problem

$$D = \{\langle 1, 01 \rangle, \langle 0, 01 \rangle, \langle 010, \varepsilon \rangle\}$$



$$\begin{array}{cccc} 0 & 1 & 010 & 1 \\ 01 & 01 & \varepsilon & 01 \end{array}$$