Beschreibungslogik | Übung 04

D. Marschner, A. Mahdavi
alma@uni-bremen.de

Aufgabe 1)

tbd

Aufgabe 2)

a)

$C_0 = \exists r.\neg A$ erfüllbar bzgl. $\mathcal{T} = \{\forall r.A \sqsubseteq A, A \sqsubseteq \bot, \forall r.A \sqsubseteq \exists r.A\}$

$\mathcal{T}$ in NNF bringen:

$$\begin{aligned}
\mathcal{T} &= \{\forall r.A \sqsubseteq A, A \sqsubseteq \bot, \forall r.A \sqsubseteq \exists r.A\} \\
&= \{\top \sqsubseteq (\neg\forall r.A \sqcup A) \sqcap (\neg A \sqcup \bot) \sqcap (\neg\forall r.A \sqcup \exists r.A)\} \\
&= \{\top \sqsubseteq (\exists r.\neg A \sqcup A) \sqcap \neg A \sqcap (\exists r.\neg A \sqcup \exists r.A)\}
\end{aligned}$$

$\mathcal{T} = \{\top \sqsubseteq C_{\mathcal{T}}\}$ mit $C_{\mathcal{T}} = \{(\exists r.\neg A \sqcup A) \sqcap \neg A \sqcap (\exists r.\neg A \sqcup \exists r.A)\}$

$sub(C_0, \mathcal{T})$ generieren:

$$sub(C_0, \mathcal{T}) = \{\exists r.\neg A, C_{\mathcal{T}}, \exists r.\neg A \sqcup A, \neg A, \exists r.\neg A \sqcup \exists r.A, A, \exists r.A\}$$

Wegen $C_{\mathcal{T}} \in t$ für jeden Typen t für C_0 und $\mathcal{T}$ und der Typ-Bedingung für $\sqcap$, muss jeder Typ die Menge $M = \{C_{\mathcal{T}}, \exists r.\neg A \sqcup A, \neg A, \exists r.\neg A \sqcup \exists r.A\}$ enthalten. Aufgrund der Regel-(1) von Definition 5.2 (Typ) und weil $\neg A \in sub(C_0, \mathcal{T})$ ist $A \notin t$. Dadurch ergibt sich mit der $\sqcup$-Regel, dass $\exists r.\neg A \in t$ sein muss. Somit ist $M = \{C_{\mathcal{T}}, \exists r.\neg A \sqcup A, \neg A, \exists r.\neg A \sqcup \exists r.A, \exists r.\neg A\}$.

Man kann sich also leicht überzeugen, dass es insgesamt zwei Typen für C_0 und $\mathcal{T}$ gibt, nämlich:

$$\begin{aligned} t_0 &= M \cup \{\exists r.A\} \\ t_1 &= M \end{aligned}$$

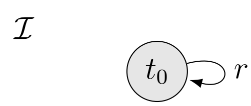
Der Typ t_0 ist schlecht in der Menge aller Typen: für $\exists r.A \in t_0$ und $\exists r.\neg A \in t_0$ ist die Menge aus Definition 5.3 $\{A, \neg A\}$, aber kein Typ enthält sowohl A als auch $\neg A$.
Der Typ t_1 ist nicht schlecht in der Menge aller Typen: für $\exists r.\neg A \in t_1$ ist die Menge aus Definition 5.3 $\{\neg A\}$, wobei t_1 selbst $\neg A$ enthält. Also $\neg A \in t_1 = t'$.

Das Typeliminationsverfahren berechnet folgende Mengen:

$$\begin{aligned} \Gamma_0 &= \{t_0, t_1\} \\ \Gamma_1 &= \{t_1\} \\ \Gamma_2 &= \{t_1\} \end{aligned}$$

Der Algorithmus stoppt, weil $\Gamma_1 = \Gamma_2$.
Das Ergebnis ist $erfüllbar$, weil es ein $t = t_1 \in \Gamma_2$ gibt mit $C_0 = \exists r.\neg A \in t$.

Model $\mathcal{I}$ aus Beweis von Lemma 5.5:



$$\begin{aligned} \Delta^{\mathcal{I}} &= \{t_0\} \\ r^{\mathcal{I}} &= \{(t_0, t_0)\} \end{aligned}$$

Da $(C_0)^{\mathcal{I}} = (\exists r.\neg A)^{\mathcal{I}} = \{t_0\} \neq \emptyset$ ist $\mathcal{I}$ Modell von C_0.

b)

$C_0 = \forall r.\forall r.A$ erfüllbar bzgl. $\mathcal{T} = \{\neg A \sqsubseteq B, A \sqsubseteq \neg B, \forall r.A \sqsubseteq \bot\}$

$\mathcal{T}$ in NNF bringen:

$$\begin{aligned}\mathcal{T} &= \{\neg A \sqsubseteq B, A \sqsubseteq \neg B, \forall r.A \sqsubseteq \bot\}\\ &= \{\top \sqsubseteq (\neg\neg A \sqcup B) \sqcap (\neg A \sqcup \neg B) \sqcap (\neg\forall r.A \sqcup \bot)\}\\ &= \{\top \sqsubseteq (A \sqcup B) \sqcap (\neg A \sqcup \neg B) \sqcap (\exists r.\neg A \sqcup \bot)\}\\ &= \{\top \sqsubseteq (A \sqcup B) \sqcap (\neg A \sqcup \neg B) \sqcap \exists r.\neg A\}\end{aligned}$$

$\mathcal{T} = \{\top \sqsubseteq C_{\mathcal{T}}\}$ mit $C_{\mathcal{T}} = \{(A \sqcup B) \sqcap (\neg A \sqcup \neg B) \sqcap \exists r.\neg A\}$

$sub(C_0, \mathcal{T})$ generieren:

$$sub(C_0, \mathcal{T}) = \{\forall r.\forall r.A, C_{\mathcal{T}}, A \sqcup B, \neg A \sqcup \neg B, \exists r.\neg A, A, B, \neg A, \neg B, \forall r.A\}$$

Wegen $C_{\mathcal{T}} \in t$ für jeden Typen t für C_0 und $\mathcal{T}$ und der Typ-Bedingung für $\sqcap$, muss jeder Typ die Menge $M = \{C_{\mathcal{T}}, A \sqcup B, \neg A \sqcup \neg B, \exists r.\neg A\}$ enthalten. Aufgrund der Regel-(1) von Definition 5.2 (Typ) und der $\sqcup$-Regel kann man sich leicht überzeugen, dass es insgesamt vier Typen für C_0 und $\mathcal{T}$ gibt, nämlich:

$$\begin{aligned}t_0 &= M \cup \{A, \neg B\}\\ t_1 &= M \cup \{\neg A, B\}\\ t_2 &= M \cup \{A, \neg B, \forall r.A\}\\ t_3 &= M \cup \{\neg A, B, \forall r.A\}\end{aligned}$$

Der Typ t_0 ist nicht schlecht in der Menge aller Typen: für $\exists r.\neg A \in t_0$ ist die Menge aus Definition 5.3 $\{\neg A\}$, wobei $\neg A \in t_1$. Also $\neg A \in t_1 = t'$.

Der Typ t_1 ist nicht schlecht in der Menge aller Typen: für $\exists r.\neg A \in t_1$ ist die Menge aus Definition 5.3 $\{\neg A\}$, wobei t_1 selbst $\neg A$ enthält. Also $\neg A \in t_1 = t'$.

Der Typ t_2 ist schlecht in der Menge aller Typen: für $\exists r.\neg A \in t_2$ und $\forall r.A \in t_2$ ist die Menge aus Definition 5.3 $\{\neg A, A\}$, aber kein Typ enthält sowohl A als auch $\neg A$.

Der Typ t_3 ist schlecht in der Menge aller Typen: für $\exists r.\neg A \in t_3$ und $\forall r.A \in t_3$ ist die Menge aus Definition 5.3 $\{\neg A, A\}$, aber kein Typ enthält sowohl A als auch $\neg A$.

Das Typeliminationsverfahren berechnet folgende Mengen:

$$\Gamma_0 = \{t_0, t_1, t_2, t_3\}$$
$$\Gamma_1 = \{t_0, t_1\}$$
$$\Gamma_2 = \{t_0, t_1\}$$

Der Algorithmus stoppt, weil $\Gamma_1 = \Gamma_2$.
Das Ergebnis ist $unerf\ddot{u}llbar$, weil es ein kein $t \in \Gamma_2$ gibt mit $C_0 = \forall r.\forall r.A \in t$.

Aufgabe 3)

tbd

Aufgabe 4)

tbd

Aufgabe 5)

tbd