Klauzulární axiomatický systém

Šárka Vavrečková

Ústav informatiky, Filozoficko-přírodovědecká fakulta Slezské univerzity v Opavě

sarka.vavreckova@fpf.slu.cz

26. listopadu 2007

Jazyk

přejímáme jazyk klauzulární logiky

Logické axiomy (A)

jsou klauzule klauzulární logiky, ve kterých se tentýž atom (včetně argumentů) vyskytuje v antecedentu i v konsekventu, tedy formule

$$p_1, p_2, \ldots, p_n, p \rightarrow p, q_1, q_2, \ldots, q_m$$

Speciální axiomy (SA)

jsou klauzule znalostní báze. Znalostní báze nesmí být sporná množina.

Jazyk

přejímáme jazyk klauzulární logiky

Logické axiomy (A)

jsou klauzule klauzulární logiky, ve kterých se tentýž atom (včetně argumentů) vyskytuje v antecedentu i v konsekventu, tedy formule

$$p_1, p_2, \ldots, p_n, p \rightarrow p, q_1, q_2, \ldots, q_m$$

Speciální axiomy (SA)

jsou klauzule znalostní báze. Znalostní báze nesmí být sporná množina.

Jazyk

přejímáme jazyk klauzulární logiky

Logické axiomy (A)

jsou klauzule klauzulární logiky, ve kterých se tentýž atom (včetně argumentů) vyskytuje v antecedentu i v konsekventu, tedy formule

$$p_1, p_2, \ldots, p_n, p \rightarrow p, q_1, q_2, \ldots, q_m$$

Speciální axiomy (SA)

jsou klauzule znalostní báze. Znalostní báze nesmí být sporná množina.

Odvozovací pravidla

1. *Odvozovací pravidlo substituce* (S):
   Jestliže $\varphi = \{t_1/X_1, t_2/X_2, \dots, t_n/X_n\}$ je substituce a $\mathcal{C}$ je klauzule, pak platí
   $$\mathcal{C} \;\vdash\; \mathcal{C}[\varphi]$$

2. *Rezoluční odvozovací pravidlo* (odvozovací pravidlo rezolučního řezu, R):
   Označme p atom a A_1, A_2, K_1, K_2 množiny atomů. Pak platí
   $$A_1, p \to K_1, \quad A_2 \to p, K_2 \quad \vdash \quad A_1, A_2 \to K_1, K_2$$

3. *Odvozovací pravidlo existenční substituce* (ES):
   Jestliže $\varphi = (@c/a; k_1, k_2, \dots, k_n)$ je existenční substituce a $\mathcal{C}$ je klauzule, pak platí
   $$\mathcal{C} \;\vdash\; \mathcal{C}[\varphi]$$

Odvozovací pravidla

1. *Odvozovací pravidlo substituce* (S):
   Jestliže $\varphi = \{t_1/X_1, t_2/X_2, \ldots, t_n/X_n\}$ je substituce a $\mathcal{C}$ je klauzule, pak platí
   $$\mathcal{C} \ \vdash \ \mathcal{C}[\varphi]$$
2. *Rezoluční odvozovací pravidlo* (odvozovací pravidlo rezolučního řezu, R):
   Označme p atom a A_1, A_2, K_1, K_2 množiny atomů. Pak platí
   $$A_1, p \rightarrow K_1, \quad A_2 \rightarrow p, K_2 \quad \vdash \quad A_1, A_2 \rightarrow K_1, K_2$$
3. *Odvozovací pravidlo existenční substituce* (ES):
   Jestliže $\varphi = (@c/a; k_1, k_2, \ldots, k_n)$ je existenční substituce a $\mathcal{C}$ je klauzule, pak platí
   $$\mathcal{C} \ \vdash \ \mathcal{C}[\varphi]$$

Odvozovací pravidla

1. *Odvozovací pravidlo substituce* (S):
Jestliže $\varphi = \{t_1/X_1, t_2/X_2, \ldots, t_n/X_n\}$ je substituce a $\mathcal{C}$ je klauzule, pak platí
$$\mathcal{C} \ \vdash \ \mathcal{C}[\varphi]$$
2. *Rezoluční odvozovací pravidlo* (odvozovací pravidlo rezolučního řezu, R):
Označme p atom a A_1, A_2, K_1, K_2 množiny atomů. Pak platí
$$A_1, p \rightarrow K_1, \quad A_2 \rightarrow p, K_2 \quad \vdash \quad A_1, A_2 \rightarrow K_1, K_2$$
3. *Odvozovací pravidlo existenční substituce* (ES):
Jestliže $\varphi = (@c/a; k_1, k_2, \ldots, k_n)$ je existenční substituce a $\mathcal{C}$ je klauzule, pak platí
$$\mathcal{C} \ \vdash \ \mathcal{C}[\varphi]$$

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ bázová substituce (BS)
     $(A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ bázová substituce (BS)
     $(A(t), A(r)$ jsou bázové atomy$)$
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \vdash p, A \rightarrow K$
   - $A \rightarrow p, p, K \vdash A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\to t = t$ — reflexivita predikátu rovnosti (RR)
   - $t = r \to r = t$ — symetrie predikátu rovnosti (SR)
   - $t = r, r = s \to t = s$ — tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \to A(r)$ — bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \to K \quad \vdash \quad p, A \to K$
   - $A \to p, p, K \quad \vdash \quad A \to p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ — reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ — symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ — tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ — bázová substituce (BS)
     $(A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ — reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ — symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ — tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ — bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ — reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ — symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ — tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ — bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)
2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$
3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Pomocná odvozovací pravidla

1. *Logické zákony pro predikát rovnosti:*
   Jestliže t, r, s jsou termy a A je predikát, pak platí
   - $\rightarrow t = t$ reflexivita predikátu rovnosti (RR)
   - $t = r \rightarrow r = t$ symetrie predikátu rovnosti (SR)
   - $t = r, r = s \rightarrow t = s$ tranzitivita predikátu rovnosti (TR)
   - $t = r, A(t) \rightarrow A(r)$ bázová substituce (BS)
     ($A(t), A(r)$ jsou bázové atomy)

2. *Pomocné odvozovací pravidlo kontrakce* (KK) – lze rovněž použít na odstranění atomu $true$ z antecedentu nebo $false$ z konsekventu:
   - $p, p, A \rightarrow K \quad \vdash \quad p, A \rightarrow K$
   - $A \rightarrow p, p, K \quad \vdash \quad A \rightarrow p, K$

3. *Pomocné odvozovací pravidlo přeuspořádání atomů* (PA):
   Atomy v antecedentu lze přeuspořádat (změnit jejich pořadí).
   Atomy v konsekventu lze přeuspořádat.

Příklad

Odvoďte pravidla *Modus Ponens* (MP) a *Modus Tolens* (MT).

$$\mathrm{MP}: \quad \rightarrow P\,, \quad P \rightarrow Q \quad \vdash \quad \rightarrow Q$$
$$\mathrm{MT}: \quad Q \rightarrow\,, \quad P \rightarrow Q \quad \vdash \quad P \rightarrow$$

1. $\rightarrow P$ $\quad SA_1$
2. $P \rightarrow Q$ $\quad SA_2$
3. $\rightarrow Q$ $\quad$ R(1,2)

1. $Q \rightarrow$ $\quad SA_1$
2. $P \rightarrow Q$ $\quad SA_2$
3. $P \rightarrow$ $\quad$ R(1,2)

Příklad

Odvoďte pravidla *Modus Ponens* (MP) a *Modus Tolens* (MT).

$$\begin{array}{lcccc} \mathrm{MP}: & \rightarrow P\ , & P \rightarrow Q & \vdash & \rightarrow Q \\ \mathrm{MT}: & Q \rightarrow\ , & P \rightarrow Q & \vdash & P \rightarrow \end{array}$$

1. $\rightarrow P$ $\quad SA_1$
2. $P \rightarrow Q$ $\quad SA_2$
3. $\rightarrow Q$ $\quad$ R(1,2)

1. $Q \rightarrow$ $\quad SA_1$
2. $P \rightarrow Q$ $\quad SA_2$
3. $P \rightarrow$ $\quad$ R(1,2)

Příklad

Odvoďte pravidla *Modus Ponens* (MP) a *Modus Tolens* (MT).

$$\begin{array}{lcccc} \mathrm{MP}: & \to P\ , & P \to Q & \vdash & \to Q \\ \mathrm{MT}: & Q \to\ , & P \to Q & \vdash & P \to \end{array}$$

1. $\to P$ $\quad SA_1$
2. $P \to Q$ $\quad SA_2$
3. $\to Q$ $\quad$ R(1,2)

1. $Q \to$ $\quad SA_1$
2. $P \to Q$ $\quad SA_2$
3. $P \to$ $\quad$ R(1,2)

Definice

Přímý důkaz klauzule $\mathcal{C}$ z ze znalostní báze (množiny speciálních axiomů) $\mathcal{B}$ je posloupnost klauzulí $A_1, A_2, \ldots, A_m$, kde $\mathcal{C} = A_m$ a pro každé $i \in \{1, \ldots, m\}$ platí pro A_i některá z těchto možností:

- $A_i \in \mathcal{B}$ *(tj. je to některý ze speciálních axiomů báze),*
- A_i *je logický axiom,*
- A_i *vznikla použitím některého odvozovacího pravidla na předchozí členy posloupnosti.*

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ $\qquad SA_1$
2. $lezi(nemecko, za\_morem) \rightarrow$ $\qquad SA_2$
3. $\rightarrow lezi(anglie, za\_morem)$ $\qquad SA_3$
4. $\rightarrow cestuje(honza, anglie)$ $\qquad SA_4$
5. $jede(honza, letadlo) \rightarrow$ $\qquad SA_5$

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ $\qquad SA_1$
2. $lezi(nemecko, za\_morem) \rightarrow$ $\qquad SA_2$
3. $\rightarrow lezi(anglie, za\_morem)$ $\qquad SA_3$
4. $\rightarrow cestuje(honza, anglie)$ $\qquad SA_4$
5. $jede(honza, letadlo) \rightarrow$ $\qquad SA_5$

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ $\qquad SA_1$
2. $lezi(nemecko, za\_morem) \rightarrow$ $\qquad SA_2$
3. $\rightarrow lezi(anglie, za\_morem)$ $\qquad SA_3$
4. $\rightarrow cestuje(honza, anglie)$ $\qquad SA_4$
5. $jede(honza, letadlo) \rightarrow$ $\qquad SA_5$

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?" *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ $\quad SA_1$
2. $lezi(nemecko, za\_morem) \rightarrow$ $\quad SA_2$
3. $\rightarrow lezi(anglie, za\_morem)$ $\quad SA_3$
4. $\rightarrow cestuje(honza, anglie)$ $\quad SA_4$
5. $jede(honza, letadlo) \rightarrow$ $\quad SA_5$

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

- Když země leží za mořem, každý, kdo do ní cestuje, jede lodí nebo letadlem.
- Německo neleží za mořem, ale Anglie ano.
- Honza cestuje do Anglie.
- Honza nejezdí letadlem.

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ $\quad SA_1$
2. $lezi(nemecko, za\_morem) \rightarrow$ $\quad SA_2$
3. $\rightarrow lezi(anglie, za\_morem)$ $\quad SA_3$
4. $\rightarrow cestuje(honza, anglie)$ $\quad SA_4$
5. $jede(honza, letadlo) \rightarrow$ $\quad SA_5$

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
7. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,6)
8. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,7)
9. $\rightarrow jede(honza, lod)$ R(5,8)

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?" *přímým důkazem*

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
7. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,6)
8. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,7)
9. $\rightarrow jede(honza, lod)$ R(5,8)

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
7. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,6)
8. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,7)
9. $\rightarrow jede(honza, lod)$ R(5,8)

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
7. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,6)
8. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,7)
9. $\rightarrow jede(honza, lod)$ R(5,8)

Příklad

Odvoďte odpověď na dotaz „Jede Honza lodí?“ *přímým důkazem*

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow$
   $jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5

6. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow$
   $jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
7. $cestuje(honza, anglie) \rightarrow jede(honza, lod),$
   $jede(honza, letadlo)$ R(3,6)
8. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,7)
9. $\rightarrow jede(honza, lod)$ R(5,8)

Definice

Nepřímý důkaz klauzule $\mathcal{C}$ ze znalostní báze $\mathcal{B} = \{SA_1, SA_2, \ldots, SA_n\}$ je posloupnost klauzulí $A_1, A_2, \ldots, A_m$, kde $A_m = \rightarrow$ (prázdná klauzule, vždy interpretovná jako $false$) a pro každé $i \in \{1, \ldots, m\}$ platí pro A_i některá z těchto možností:

- *A_i patří do popírající množiny klauzule $\mathcal{C}$,*
- *$A_i \in \mathcal{B}$ (tj. je to některý ze speciálních axiomů báze),*
- *A_i je logický axiom,*
- *A_i vznikla použitím některého odvozovacího pravidla na předchozí členy posloupnosti.*

Postup

- vytvoříme *popírající množinu* pro klauzuli, kterou chceme dokázat,
- tuto popírající množinu (PM) přidáme k znalostní bázi,
- odvozujeme s použitím odvozovacích pravidel,
- jestliže odvodíme prázdnou klauzuli ($\rightarrow$), klauzule je dokázána (vyplývá ze znalostní báze).

Postup

- vytvoříme *popírající množinu* pro klauzuli, kterou chceme dokázat,
- tuto popírající množinu (PM) přidáme k znalostní bázi,
- odvozujeme s použitím odvozovacích pravidel,
- jestliže odvodíme prázdnou klauzuli ($\rightarrow$), klauzule je dokázána (vyplývá ze znalostní báze).

Postup

- vytvoříme *popírající množinu* pro klauzuli, kterou chceme dokázat,
- tuto popírající množinu (PM) přidáme k znalostní bázi,
- odvozujeme s použitím odvozovacích pravidel,
- jestliže odvodíme prázdnou klauzuli ($\rightarrow$), klauzule je dokázána (vyplývá ze znalostní báze).

Postup

- vytvoříme *popírající množinu* pro klauzuli, kterou chceme dokázat,
- tuto popírající množinu (PM) přidáme k znalostní bázi,
- odvozujeme s použitím odvozovacích pravidel,
- jestliže odvodíme prázdnou klauzuli ($\rightarrow$), klauzule je dokázána (vyplývá ze znalostní báze).

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ — SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ — SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ — SA_3
4. $\rightarrow cestuje(honza, anglie)$ — SA_4
5. $jede(honza, letadlo) \rightarrow$ — SA_5
6. $jede(honza, lod) \rightarrow$ — PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(4,8)
10. $\rightarrow jede(honza, lod)$ — R(5,9)
11. $\rightarrow$ — R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow$
   $jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $jede(honza, lod) \rightarrow$ PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow$
   $jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod),$
   $jede(honza, letadlo)$ R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,8)
10. $\rightarrow jede(honza, lod)$ R(5,9)
11. $\rightarrow$ R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $jede(honza, lod) \rightarrow$ PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,8)
10. $\rightarrow jede(honza, lod)$ R(5,9)
11. $\rightarrow$ R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ SA_3
4. $\rightarrow cestuje(honza, anglie)$ SA_4
5. $jede(honza, letadlo) \rightarrow$ SA_5
6. $jede(honza, lod) \rightarrow$ PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ R(4,8)
10. $\rightarrow jede(honza, lod)$ R(5,9)
11. $\rightarrow$ R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ — SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ — SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ — SA_3
4. $\rightarrow cestuje(honza, anglie)$ — SA_4
5. $jede(honza, letadlo) \rightarrow$ — SA_5
6. $jede(honza, lod) \rightarrow$ — PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(4,8)
10. $\rightarrow jede(honza, lod)$ — R(5,9)
11. $\rightarrow$ — R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ — SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ — SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ — SA_3
4. $\rightarrow cestuje(honza, anglie)$ — SA_4
5. $jede(honza, letadlo) \rightarrow$ — SA_5
6. $jede(honza, lod) \rightarrow$ — PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(4,8)
10. $\rightarrow jede(honza, lod)$ — R(5,9)
11. $\rightarrow$ — R(6,10)

1. $lezi(X, za\_morem), cestuje(Y, X) \rightarrow jede(Y, lod), jede(Y, letadlo)$ — SA_1
2. $lezi(nemecko, za\_morem) \rightarrow$ — SA_2
3. $\rightarrow lezi(anglie, za\_morem)$ — SA_3
4. $\rightarrow cestuje(honza, anglie)$ — SA_4
5. $jede(honza, letadlo) \rightarrow$ — SA_5
6. $jede(honza, lod) \rightarrow$ — PM
7. $lezi(anglie, za\_morem), cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ S(1)$\{anglie/X, honza/Y\}$
8. $cestuje(honza, anglie) \rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(3,7)
9. $\rightarrow jede(honza, lod), jede(honza, letadlo)$ — R(4,8)
10. $\rightarrow jede(honza, lod)$ — R(5,9)
11. $\rightarrow$ — R(6,10)

Věta o korektnosti KAS

Věta

Klauzulární axiomatický systém je korektní.

Důkaz věty o korektnosti KAS

1) Logické axiomy

Logický axiom je každá klauzule, jejíž množiny antecedentu a konsekventu mají neprázdný průnik:

$$A, p \ \rightarrow p, K$$

Po převodu do predikátové logiky:

$$(A \,\&\, p) \rightarrow (p \vee K) \ \Leftrightarrow\ (\neg A \vee \neg p) \vee (p \vee K) \ \Leftrightarrow$$
$$\neg A \vee (\neg p \vee p) \vee K \ \Leftrightarrow\ \neg A \vee 1 \vee K \ \Leftrightarrow\ 1$$

$\Longrightarrow$ každý logický axiom je logicky platná formule. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

1) Logické axiomy

Logický axiom je každá klauzule, jejíž množiny antecedentu a konsekventu mají neprázdný průnik:

$$A, p \ \rightarrow p, K$$

Po převodu do predikátové logiky:

$$(A \,\&\, p) \rightarrow (p \vee K) \ \Leftrightarrow \ (\neg A \vee \neg p) \vee (p \vee K) \ \Leftrightarrow \\ \neg A \vee (\neg p \vee p) \vee K \ \Leftrightarrow \ \neg A \vee 1 \vee K \ \Leftrightarrow \ 1$$

$\Longrightarrow$ každý logický axiom je logicky platná formule. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

1) Logické axiomy

Logický axiom je každá klauzule, jejíž množiny antecedentu a konsekventu mají neprázdný průnik:

$$A, p \ \rightarrow p, K$$

Po převodu do predikátové logiky:

$$(A \,\&\, p) \rightarrow (p \vee K) \ \Leftrightarrow\ (\neg A \vee \neg p) \vee (p \vee K) \ \Leftrightarrow \\ \neg A \vee (\neg p \vee p) \vee K \ \Leftrightarrow\ \neg A \vee 1 \vee K \ \Leftrightarrow\ 1$$

$\Longrightarrow$ každý logický axiom je logicky platná formule. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

2) Odvozovací pravidla

- Rezoluční odvozovací pravidlo – důkaz proveden dříve. $\Rightarrow$ OK
- Odvozovací pravidlo substituce – založeno na syntaxi predikátové logiky, dokázáno. $\Rightarrow$ OK
- Odvozovací pravidlo existenční substituce – vyplývá z vlastností existenčně vázaných proměnných. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

2) Odvozovací pravidla

- Rezoluční odvozovací pravidlo – důkaz proveden dříve. $\Rightarrow$ OK
- Odvozovací pravidlo substituce – založeno na syntaxi predikátové logiky, dokázáno. $\Rightarrow$ OK
- Odvozovací pravidlo existenční substituce – vyplývá z vlastností existenčně vázaných proměnných. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

2) Odvozovací pravidla

- Rezoluční odvozovací pravidlo – důkaz proveden dříve. $\Rightarrow$ OK
- Odvozovací pravidlo substituce – založeno na syntaxi predikátové logiky, dokázáno. $\Rightarrow$ OK
- Odvozovací pravidlo existenční substituce – vyplývá z vlastností existenčně vázaných proměnných. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

3) Korektnost posloupnosti důkazu

- báze: prvním prvkem posloupnosti důkazu je logický axiom (to je logicky platná formule) nebo logický axiom (ten jen omezuje množinu valuací, ve kterých musí být závěr splnitelný $\Rightarrow$ OK
- předpoklad indukce: důkaz je proveden pro prvních k prvků posloupnosti důkazu.
- krok indukce: pro $k+1$. prvek posloupnosti důkazu platí:
  - logický nebo speciální axiom – stejně jako v bázi; $\Rightarrow$ OK
  - vznikl použitím rezolučního pravidla, pravidla substituce nebo existenční substituce na některý z předchozích členů posloupnosti s indexem $< (k+1)$ – pro tyto předchozí členy již je věta dokázána, a máme dokázáno, že pravidla zachovávají splnitelnost – $k+1$. člen důkazu má tutéž vlastnost. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

3) Korektnost posloupnosti důkazu

- báze: prvním prvkem posloupnosti důkazu je logický axiom (to je logicky platná formule) nebo logický axiom (ten jen omezuje množinu valuací, ve kterých musí být závěr splnitelný $\Rightarrow$ OK
- předpoklad indukce: důkaz je proveden pro prvních k prvků posloupnosti důkazu.
- krok indukce: pro $k+1$. prvek posloupnosti důkazu platí:
  - logický nebo speciální axiom – stejně jako v bázi; $\Rightarrow$ OK
  - vznikl použitím rezolučního pravidla, pravidla substituce nebo existenční substituce na některý z předchozích členů posloupnosti s indexem $< (k+1)$ – pro tyto předchozí členy již je věta dokázána, a máme dokázáno, že pravidla zachovávají splnitelnost – $k+1$. člen důkazu má tutéž vlastnost. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

3) Korektnost posloupnosti důkazu

- báze: prvním prvkem posloupnosti důkazu je logický axiom (to je logicky platná formule) nebo logický axiom (ten jen omezuje množinu valuací, ve kterých musí být závěr splnitelný $\Rightarrow$ OK
- předpoklad indukce: důkaz je proveden pro prvních k prvků posloupnosti důkazu.
- krok indukce: pro $k+1$. prvek posloupnosti důkazu platí:
  - logický nebo speciální axiom – stejně jako v bázi; $\Rightarrow$ OK
  - vznikl použitím rezolučního pravidla, pravidla substituce nebo existenční substituce na některý z předchozích členů posloupnosti s indexem $< (k+1)$ – pro tyto předchozí členy již je věta dokázána, a máme dokázáno, že pravidla zachovávají splnitelnost – $k+1$. člen důkazu má tutéž vlastnost. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

3) Korektnost posloupnosti důkazu

- báze: prvním prvkem posloupnosti důkazu je logický axiom (to je logicky platná formule) nebo logický axiom (ten jen omezuje množinu valuací, ve kterých musí být závěr splnitelný $\Rightarrow$ OK
- předpoklad indukce: důkaz je proveden pro prvních k prvků posloupnosti důkazu.
- krok indukce: pro $k+1$. prvek posloupnosti důkazu platí:
  - logický nebo speciální axiom – stejně jako v bázi; $\Rightarrow$ OK
  - vznikl použitím rezolučního pravidla, pravidla substituce nebo existenční substituce na některý z předchozích členů posloupnosti s indexem $< (k+1)$ – pro tyto předchozí členy již je věta dokázána, a máme dokázáno, že pravidla zachovávají splnitelnost – $k+1$. člen důkazu má tutéž vlastnost. $\Rightarrow$ OK

Důkaz věty o korektnosti KAS

3) Korektnost posloupnosti důkazu

- báze: prvním prvkem posloupnosti důkazu je logický axiom (to je logicky platná formule) nebo logický axiom (ten jen omezuje množinu valuací, ve kterých musí být závěr splnitelný $\Rightarrow$ OK
- předpoklad indukce: důkaz je proveden pro prvních k prvků posloupnosti důkazu.
- krok indukce: pro $k+1$. prvek posloupnosti důkazu platí:
  - logický nebo speciální axiom – stejně jako v bázi; $\Rightarrow$ OK
  - vznikl použitím rezolučního pravidla, pravidla substituce nebo existenční substituce na některý z předchozích členů posloupnosti s indexem $< (k+1)$ – pro tyto předchozí členy již je věta dokázána, a máme dokázáno, že pravidla zachovávají splnitelnost – $k+1$. člen důkazu má tutéž vlastnost. $\Rightarrow$ OK