METODY DOWODZENIA TWIERDZEŃ I AUTOMATYZACJA ROZUMOWAŃ

WYKŁAD 14: POWTÓRKA

III rok kognitywistyki UAM, 2016–2017

Dzisiejszy wykład w całości poświęcony będzie omówieniu przykładowych zadań, podobnych do tych, które pojawią się na zaliczeniu wykładu za tydzień, czyli 24 stycznia 2014.

1 Notacja

Zakładamy, że słuchacze potrafią posługiwać się notacją stosowaną w języku klasycznego rachunku zdań (KRZ) oraz klasycznego rachunku predykatów (KRP). We wszystkich zadaniach, dotyczących przeprowadzania dowodów (w rozważanym systemie dowodowym) należy poszczególne kroki dowodowe opatrywać komentarzami je uzasadniającymi (zob. przykłady poniżej).

1.1 Notacja Smullyana

Przypominamy tabelę składników α-formuł i β formuł:

α	α_1	α_2
$\varphi \wedge \psi$	φ	ψ
$\neg(\varphi \vee \psi)$	$\neg\varphi$	$\neg\psi$
$\neg(\varphi \rightarrow \psi)$	φ	$\neg\psi$
$\neg(\varphi \leftarrow \psi)$	$\neg\varphi$	ψ
$\neg(\varphi \uparrow \psi)$	φ	ψ
$\varphi \downarrow \psi$	$\neg\varphi$	$\neg\psi$
$\varphi \not\rightarrow \psi$	φ	$\neg\psi$
$\varphi \not\leftarrow \psi$	$\neg\varphi$	ψ

β	β_1	β_2
$\neg(\varphi \wedge \psi)$	$\neg\varphi$	$\neg\psi$
$\varphi \vee \psi$	φ	ψ
$\varphi \rightarrow \psi$	$\neg\varphi$	ψ
$\varphi \leftarrow \psi$	φ	$\neg\psi$
$\varphi \uparrow \psi$	$\neg\varphi$	$\neg\psi$
$\neg(\varphi \downarrow \psi)$	φ	ψ
$\neg(\varphi \not\rightarrow \psi)$	$\neg\varphi$	ψ
$\neg(\varphi \not\leftarrow \psi)$	φ	$\neg\psi$

1.2 Notacja polska

Zakładamy, że słuchacze dobrze znają notację *infiksową* (symbol funktora dwuargumentowego pomiędzy symbolami jego argumentów). Reguły składni wymagają wtedy korzystania z nawiasów. W notacji *prefiksowej* (nazywanej też notacją *polską* lub notacją *Łukasiewicza*) symbol funktora poprzedza symbole swoich argumentów, nawiasy stają się zbędne. Oczywiście przyjąć trzeba ustalone symbole dla funktorów różne od symboli dla zmiennych, np.:

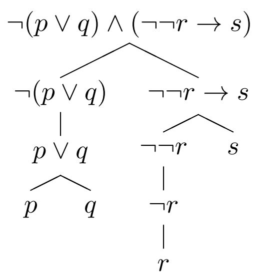
1. N – negacja
2. A – alternatywa
3. K – koniunkcja
4. C – implikacja
5. E – równoważność.

Dla przykładu:

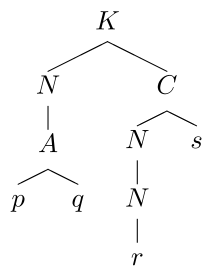
1. Formuła w postaci infiksowej $((p \to q) \wedge (p \to \neg q)) \to q$ przekształca się na formułę $CKCpqCpNqq$ w postaci prefiksowej.
2. Formuła $KNApqCNNrs$ w postaci prefiksowej przekształca się na formułę $\neg(p \vee q) \wedge (\neg\neg r \to s)$ w postaci infiksowej.

Każda z tych notacji przekłada się również na reprezentację w postaci *drzew składniowych*, które powinny być znane słuchaczom z kursu *Logika I*. Drzewa składniowe możemy zapisywać w postaci pełnej (wyraźnie podając wszystkie podformuły rozważanej formuły) lub w postaci skróconej (umieszczając w liściach tego drzewa występujące w formule zmienne zdaniowe i znakując pozostałe wierzchołki funktorami głównymi kolejnych podformuł rozważanej formuły).

Pełne drzewo składniowe formuły $\neg(p \vee q) \wedge (\neg\neg r \to s)$ (w notacji infiksowej) wygląda następująco:



Skrócone drzewo składniowe tej samej formuły (w notacji prefiksowej, czyli $KNApqCNNrs$) wygląda następująco:

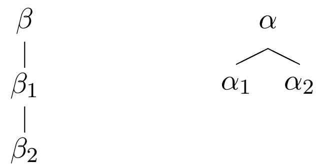


Proponujemy jako zadanie zaliczeniowe: przekształcić formułę zapisaną w postaci infiksowej na odpowiadającą jej formułę w postaci prefiksowej i zbudować drzewo składniowe rozważanej formuły (pełne lub skrócone).

2 Postacie normalne

Wymagana jest umiejętność zastosowania algorytmu znajdowania koniunkcyjnej postaci normalnej formuły języka KRZ. Korzystamy z notacji Smullyana. Reguły redukcji (Fitting 1990, 26), wykorzystywane w tworzeniu kpn:

$$\frac{\neg\neg\psi}{\psi} \qquad \frac{\neg\top}{\bot} \qquad \frac{\neg\bot}{\top}$$



Aby sprowadzić φ do kpn (Fitting 1990, 27) korzystamy z algorytmu:

1. **begin**
   (a) Niech S będzie $\langle[\varphi]\rangle$
   (b) **while** *jakiś element S zawiera nie-literał* **do**
      i. wybierz z S element D zawierający nie-literał
      ii. wybierz z D nie-literał N
      iii. zastosuj odpowiednią regułę redukcyjną do N
      iv. niech S oznacza nowoutworzoną formułę
   (c) **end**
2. **end**

Postępowanie wedle tego algorytmu dostarcza kpn dla φ. Proponujemy rozumieć zapis $\langle[\varphi]\rangle$ jako polecenie wykonania obliczenia wedle wspomnianego algorytmu dla argumentu φ.

Przykład: kpn dla $(p \to (q \to r)) \to ((p \to q) \to (p \to r))$.

1. $\langle[\underline{(p \to (q \to r)) \to ((p \to q) \to (p \to r))}]\rangle$
2. $\langle[\neg(p \to (q \to r)), \underline{((p \to q) \to (p \to r))}]\rangle$
3. $\langle[\neg(p \to (q \to r)), \neg(p \to q), \underline{(p \to r)}]\rangle$
4. $\langle[\underline{\neg(p \to (q \to r))}, \neg(p \to q), \neg p, r]\rangle$
5. $\langle[p, \underline{\neg(p \to q)}, \neg p, r], [\neg(q \to r), \neg(p \to q), \neg p, r]\rangle$
6. $\langle[p, p, \neg p, r], [p, \neg q, \neg p, r], [\underline{\neg(q \to r)}, \neg(p \to q), \neg p, r]\rangle$
7. $\langle[p, p, \neg p, r], [p, \neg q, \neg p, r], [q, \underline{\neg(p \to q)}, \neg p, r], [\neg r, \neg(p \to q), \neg p, r]\rangle$
8. $\langle[p, p, \neg p, r], [p, \neg q, \neg p, r], [q, p, \neg p, r], [q, \neg q, \neg p, r],$
   $[\neg r, \underline{\neg(p \to q)}, \neg p, r]\rangle$
9. $\langle[p, p, \neg p, r], [p, \neg q, \neg p, r], [q, p, \neg p, r], [q, \neg q, \neg p, r],$
   $[\neg r, p, \neg p, r], [\neg r, \neg q, \neg p, r]\rangle$

Podkreślono formułę, do której stosowano regułę redukcji.

Ponieważ wykładowca nie znajduje w sobie ani krzty sadyzmu, przykładowe zadanie zaliczeniowe będzie o wiele prostsze, np. takie:

Utwórz kpn dla formuły $\neg(p \to r) \vee \neg(q \vee s)$. Działamy wedle podanego algorytmu:

$$
\begin{array}{l}
\langle[\underline{\neg(p \to r) \vee \neg(q \vee s)}]\rangle \\
\langle[\neg(p \to r), \underline{\neg(q \vee s)}]\rangle \\
\langle[\underline{\neg(p \to r)}, \neg q], [\neg(p \to r), \neg s]\rangle \\
\langle[p, \neg q], [\neg r, \neg q], [\underline{\neg(p \to r)}, \neg s]\rangle \\
\langle[p, \neg q], [\neg r, \neg q], [p, \neg s], [\neg r, \neg s]\rangle
\end{array}
$$

Tak więc kpn badanej formuły to: $(p \vee \neg q) \wedge (\neg r \vee \neg q) \wedge (p \vee \neg s) \wedge (\neg r \vee \neg s)$.

Zauważmy, że badana formuła nie jest tautologią KRZ, ponieważ nie jest tak, iżby każda alternatywa elementarna wchodząca w skład powyższej koniunkcji zawierała parę literałów komplementarnych.

3 Tablice analityczne

Przypominamy jednolitą notację dla formuł z kwantyfikatorami:

γ	$\gamma(t)$	δ	$\delta(t)$
$\forall x\varphi$	$\varphi(x/t)$	$\exists x\varphi$	$\varphi(x/t)$
$\neg\exists x\varphi$	$\neg\varphi(x/t)$	$\neg\forall x\varphi$	$\neg\varphi(x/t)$

W przypadku języków pierwszego rzędu, dodajemy następujące reguły rozszerzania tablic analitycznych (w notacji Smullyana):

$$\begin{array}{c} \gamma \\ | \\ \gamma(t) \end{array} \qquad \text{(dla dowolnego termu zamkniętego języka } L^{\mathbf{par}}\text{)}$$

$$\begin{array}{c} \delta \\ | \\ \delta(a) \end{array} \qquad \text{(dla dowolnego nowego parametru } a\text{)}$$

Wymagana jest umiejętność przeprowadzania dowodów tablicowych w KRP. Przewidujemy dwa typy zadań:

1. Zbadaj, czy dane zdanie jest tezą tablicową KRP.

2. Zbadaj, czy wniosek wynika tablicowo z przesłanek.

3.1 Przykładowe zadanie pierwszego typu

Zbadaj, czy jest tezą tablicową zdanie $\exists x\forall y\ P(x,y) \to \exists z\ P(z,z)$.

Budujemy tablicę analityczną dla formuły $\neg(\exists x\forall y\ P(x,y) \to \exists z\ P(z,z))$:

$$\begin{array}{c}
\neg(\exists x\forall y\ P(x,y) \to \exists z\ P(z,z))\ ^{1.\neg\to} \\
| \\
(1_g)\ \ \exists x\forall y\ P(x,y)\ ^{2.\checkmark a} \\
| \\
(1_d)\ \ \neg\exists z\ P(z,z)\ ^{3.\star a} \\
| \\
(2)\ \ \forall y\ P(a,y)\ ^{4.\star a} \\
| \\
(3)\ \ \neg P(a,a) \\
| \\
(4)\ \ P(a,a) \\
| \\
\times_{3,4}
\end{array}$$

Ponieważ tablica jest zamknięta, więc zdanie $\exists x \forall y\ P(x,y) \to \exists z\ P(z,z)$ jest tezą systemu tablicowego dla KRP.

Przypominamy: jeśli tablica analityczna dla zdania $\neg\varphi$ nie jest zamknięta, to φ nie jest tezą systemu tablicowego dla KRP.

3.2 Przykładowe zadania drugiego typu

1. Ustal czy zdanie $\exists x(S(x) \wedge P(x))$ wynika tablicowo ze zdania:

$$\neg\forall x(M(x) \to \neg P(x)) \wedge \neg\exists x(\neg S(x) \wedge M(x))$$

Budujemy tablicę analityczną rozpoczynającą się od przesłanki oraz zaprzeczonego wniosku:

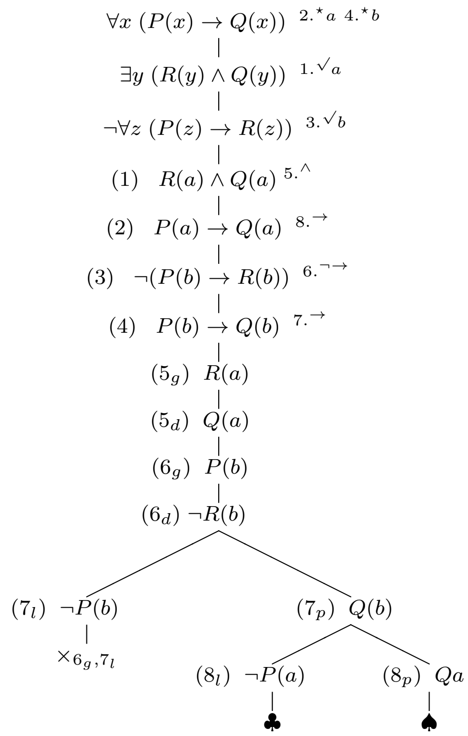
$$\neg\forall x(M(x) \to \neg P(x)) \wedge \neg\exists x(\neg S(x) \wedge M(x))\ 1.^{\wedge}$$
$$\neg\exists x(S(x) \wedge P(x))\ 3.^{*a}$$
$$(1_g)\ \neg\forall x(M(x) \to \neg P(x))\ 2.^{\surd a}$$
$$(1_d)\ \neg\exists x(\neg S(x) \wedge M(x))\ 4.^{*a}$$
$$(2)\ \neg(M(a) \to \neg P(a))\ 5.^{\neg\to}$$
$$(3)\ \neg(S(a) \wedge P(a))\ 6.^{\neg\wedge}$$
$$(4)\ \neg(\neg S(a) \wedge M(a))\ 7.^{\neg\wedge}$$
$$(5_g)\ M(a)$$
$$(5_d)\ \neg\neg P(a)$$
$$(6_l)\ \neg S(a) \qquad\qquad (6_p)\ \neg P(a)$$
$$(7_l)\ \neg\neg S(a) \qquad (7_p)\ \neg M(a) \qquad \times_{5_d,6_p}$$
$$\times_{6_l,7_l} \qquad\qquad \times_{5_g,7_p}$$

Ponieważ tablica analityczna dla przesłanki oraz zaprzeczonego wniosku jest zamknięta, więc wniosek wynika tablicowo z przesłanki.

2. Ustalimy czy wniosek wynika tablicowo z przesłanek:

$$\begin{array}{c} \forall x\ (P(x) \to Q(x)) \\ \exists y\ (R(y) \wedge Q(y)) \\ \hline \forall z\ (P(z) \to R(z)) \end{array}$$

Budujemy zatem tablicę analityczną, rozpoczynając ją od przesłanek oraz zaprzeczonego wniosku:



Tablica nie jest zamknięta, a zatem wniosek nie wynika tablicowo z przesłanek. Otwarte gałęzie tablicy pozwalają zbudować modele, w których prawdziwe są przesłanki, a fałszywy jest wniosek:

♣	P	Q	R
a	–	+	+
b	+	+	–

♠	P	Q	R
a	?	+	+
b	+	+	–

Ta notacja powinna być oczywista, dla przykładu:

1. Znak $+$ na przecięciu kolumny dla P oraz wiersza dla b w drugiej z tych tabelek oznacza, że na gałęzi ♠ znajduje się zdanie atomowe $P(b)$.

2. Znak $-$ na przecięciu kolumny dla R oraz wiersza dla b w drugiej z tych tabelek oznacza, że na gałęzi ♠ znajduje się zdanie atomowe $\neg R(b)$.

3. Znak ? na przecięciu kolumny dla P oraz wiersza dla a w drugiej z tych tabelek oznacza, że na gałęzi ♠ nie ma ani zdania atomowego $P(a)$ ani zdania atomowego $\neg P(a)$.

Zauważmy ponadto, że pierwsza przesłanka jest formułą typu γ, a więc należało zastosować do niej stosowną regułę zarówno dla stałej a, jak i dla stałej b.

4 Rezolucja

Reguły *tworzenia derywacji rezolucyjnej* zapisane w notacji Smullyana (dla języka KRZ) są następujące:

$$\frac{\neg\neg\psi}{\psi} \qquad \frac{\neg\top}{\bot} \qquad \frac{\neg\bot}{\top}$$

$$\begin{array}{c} \beta \\ | \\ \beta_1 \\ | \\ \beta_2 \end{array} \qquad\qquad \begin{array}{c} \alpha \\ \diagup\ \diagdown \\ \alpha_1 \quad \alpha_2 \end{array}$$

Reguła dla β-formuł działa wewnątrz alternatywy, reguła dla α-formuł tworzy dwie alternatywy, zapisywane jedna pod drugą.
Reguła rezolucji. Jeśli alternatywa D_1 zawiera literał L, natomiast alternatywa D_2 zawiera literał do niego komplementarny $\overline{L}$, to *rezolwentą* D_1 oraz D_2 względem L jest alternatywa D powstająca poprzez usunięcie wszystkich wystąpień L z D_1, usunięcie wszystkich wystąpień $\overline{L}$ z D_2 i utworzenie jednej alternatywy z tak otrzymanych wyników.

Wymagana jest umiejętność przeprowadzania dowodów rezolucyjnych w KRZ. Przewidujemy dwa typy zadań:

1. Wykazać, że dana formuła jest tezą rezolucyjną w KRZ.

2. Zbadać, czy dany zbiór formuł jest rezolucyjnie sprzeczny, czyli czy z danego zbioru formuł można wyprowadzić klauzulę pustą.

4.1 Przykładowe zadanie pierwszego typu

Pokaż, że dowód rezolucyjny ma formuła:

$$(p \to r) \to ((q \to r) \to ((p \vee q) \to r)).$$

Budujemy dowód rezolucyjny, czyli pokazujemy, że można wyprowadzić klauzulę pustą, wychodząc od formuły $\neg((p \to r) \to ((q \to r) \to ((p \vee q) \to r)))$:

1.	$[\neg((p \to r) \to ((q \to r) \to ((p \vee q) \to r)))]$	
2.	$[p \to r]$	α,1
3.	$[\neg((q \to r) \to ((p \vee q) \to r))]$	α,1
4.	$[q \to r]$	α,3
5.	$[\neg((p \vee q) \to r)]$	α,3
6.	$[p \vee q]$	α,5
7.	$[\neg r]$	α,5
8.	$[\neg p, r]$	β,2
9.	$[\neg q, r]$	β,4
10.	$[p, q]$	β,6
11.	$[q, r]$	RR: 8,10
12.	$[r]$	RR: 9,11
13.	$[\,]$	RR: 7,12

Uzyskano klauzulę pustą, a więc badana formuła ma dowód rezolucyjny.

4.2 Przykładowe zadanie drugiego typu

Ustal czy jest rezolucyjnie sprzeczny zbiór formuł:

$$\{\ s \wedge \neg p,\ \neg q \vee p,\ \neg r \vee q,\ \neg s \vee r\ \}$$

Pokażemy, że z podanego zbioru formuł można wyprowadzić rezolucyjnie klauzulę pustą (a więc jest on rezolucyjnie sprzeczny):

1.	$[s \wedge \neg p]$	
2.	$[\neg q \vee p]$	
3.	$[\neg r \vee q]$	
4.	$[\neg s \vee r]$	
5.	$[s]$	α,1
6.	$[\neg p]$	α,1
7.	$[\neg q, p]$	β,2
8.	$[\neg r, q]$	β,3
9.	$[\neg s, r]$	β,4
10.	$[\neg q]$	RR:6,7
11.	$[\neg r]$	RR:8,10
12.	$[\neg s]$	RR:9,11
13.	[ ]	RR:5,12

5 Dowody założeniowe

Wymagana jest umiejętność przeprowadzania dowodów założeniowych (wprost oraz nie wprost) w KRZ. Przykładowe zadania:
1. Pokaż, że z przesłanek: *Bóg jest miłosierny, o ile jest doskonały. Jeśli Bóg jest doskonały i stworzył Świat, to w Świecie nie ma Zła. Jednak w Świecie jest Zło. Ponadto, Bóg przecież stworzył Świat.* wynika logicznie wniosek: *Bóg nie jest doskonały lub nie jest miłosierny.*

Znajdujemy zdania proste:

1. p — Bóg jest doskonały.
2. q — Bóg jest miłosierny.
3. r — Bóg stworzył Świat.
4. s — W Świecie jest Zło.

Znajdujemy struktury składniowe przesłanek:

1. $p \rightarrow q$: Bóg jest miłosierny, o ile jest doskonały.
2. $(p \wedge r) \rightarrow \neg s$: Jeśli Bóg jest doskonały i stworzył Świat, to w Świecie nie ma Zła.
3. s: W Świecie jest Zło.
4. r: Bóg stworzył Świat.

Strukturą składniową wniosku *Bóg nie jest doskonały lub nie jest miłosierny* jest: $\neg p \vee \neg q$. Pokażemy, że z przesłanek wynika logicznie wniosek:

1.	$p \to q$	założenie
2.	$(p \wedge r) \to \neg s$	założenie
3.	s	założenie
4.	r	założenie
5.	$\neg\neg s$	DN: 3
6.	$\neg(p \wedge r)$	MT: 2,5
7.	$\neg p \vee \neg r$	NK: 6
8.	$\neg\neg r$	DN: 4
9.	$\neg p$	OA: 7,8
10.	$\neg p \vee \neg q$	DA: 9.

Zauważmy, że w dowodzie nie korzystano z pierwszego założenia.

1. Pokaż, że jest tezą systemu założeniowego KRZ:

$$((p \wedge q) \to r) \to ((p \wedge \neg r) \to \neg q).$$

Zbudujemy dowód założeniowy nie wprost:

1.	$(p \wedge q) \to r$	założenie
2.	$p \wedge \neg r$	założenie
3.	$\neg\neg q$	z.d.n.
4.	q	ON: 3
5.	p	OK: 2
6.	$\neg r$	OK: 2
7.	$p \wedge q$	DK: 5,4
8.	r	RO: 1,7
9.	$\perp$	sprzeczność: 6,8

Uzyskaliśmy sprzeczność przy założeniu dowodu nie wprost, a więc badana formuła jest tezą systemu założeniowego KRZ.

6 Definicje i twierdzenia

Wymagana jest znajomość następujących definicji oraz sformułowań twierdzeń:

1. Definicja zdaniowego zbioru Hintikki.

2. Definicja zdaniowej własności niesprzeczności.

3. Sformułowanie twierdzenia o dedukcji dla KRZ.

4. Sformułowanie twierdzenia o zwartości dla KRZ.

5. Sformułowanie lematu Hintikki dla KRZ.

6. Sformułowanie twierdzenia o istnieniu modelu dla KRZ.

7. Sformułowanie lematu Königa.

Wymienione definicje i twierdzenia znajdą słuchacze w tekstach zamieszczonych na stronie wykładów: http://logic.amu.edu.pl/index.php/Mdtiar

JERZY POGONOWSKI
Zakład Logiki i Kognitywistyki UAM
www.kognitywistyka.amu.edu.pl
http://logic.amu.edu.pl/index.php/Dydaktyka
pogon@amu.edu.pl