Logica predicatelor

Curs 8

„Afară plouă" $p \equiv P(a)$ a-azi
„În fiecare zi plouă afară" $(\forall x)P(x)$ x-ziua
„Mâine va ploua" $P(b)$ b-mâine
„Acum o lună a plouat" $P(f(a))$ $(f(x) := x-30)$

Sistemul axiomatic al logicii predicatelor de ordin I $\to P = (\Sigma_{P_r}, F_{P_r}, A_{P_r}, R_{P_r})$

- alfabetul

$$\Sigma_{P_r} = Var \cup Const \cup \left(\bigcup_{j=1}^{m} \mathcal{F}_j\right) \cup \left(\bigcup_{j=1}^{m} \mathcal{P}_j\right) \cup \mathcal{P}_0 \cup Conective \cup Cuantif$$

! $Var = \{x, y, z, \dots\}$ – mulțimea simbolurilor de variabile

$Const = \{a, b, c, \dots\}$ – mulțimea constantelor

! aritate = nr de termeni componenți a unei funcții

$\boxed{\text{variabilele o să se înlocuiască}}$

$\mathcal{F}_j = \{f \mid f : D^j \to D\}$ – mulțimea simbolurilor de funcții de aritate „j"

ex $f : N^2 \to N$, $f(x,y) = x^y$

$\mathcal{P}_j = \{P \mid P : D^j \to \{T, F\}\}$ → mulțimea simbolurilor de predicate de aritate „j"

ex $P : N^2 \to \{T, F\}$, $P(x,y) = $ „$x > y$"

$\mathcal{P}_0 = \{p, q, r, \dots\} \cup \{T, F\}$ – mulțimea variabilelor propoziționale și a valorilor de adevăr

$Conective = \{\neg, \vee, \wedge, \to, \leftrightarrow\}$

$Cuantif = \{\forall$ (cuantificatorul universal), $\exists$ (cuantificatorul existențial)$\}$

TERM – mulțimea termenilor

$Var \subset TERM$

$Const \subset TERM$

dacă $f \in \mathcal{F}_k$ și $t_1, \dots, t_k \in TERM$ atunci $f(t_1, \dots, t_k) \in TERM$

- mulțimea formulelor atomice (atomilor) ATOM

$T, F \in ATOM$

dacă $P \in \mathcal{P}_k$ și $t_1, \dots, t_k \in TERM$ atunci $P(t_1, \dots, t_k) \in ATOM$

- LITERAL = un atom sau negația sa

- mulțimea formulelor predicative bine formate
  - $\mathcal{F}_{Pr}$
  - ATOM $\in \mathcal{F}_{Pr}$
  - dacă $U \in \mathcal{F}_{Pr}$ și $x \in Var$ a.î. x nu se află deja sub incidența unui cuantificator (nu este legat), atunci:

$$(\forall x)\, U(x) \in \mathcal{F}_{Pr} \quad \text{și} \quad (\exists x)\, U(x) \in \mathcal{F}_{Pr}$$

  - dacă $U, V \in \mathcal{F}_{Pr}$ a.î. U și V nu conțin aceeași variabilă atât liberă cât și legată:

$$\neg U \in \mathcal{F}_{Pr},\ U \wedge V \in \mathcal{F}_{Pr},\ U \vee V \in \mathcal{F}_{Pr},\ U \to V \in \mathcal{F}_{Pr},\ U \leftrightarrow V \in \mathcal{F}_{Pr}$$

- axiome

$\mathcal{A}_{Pr} = \{A_1, A_2, A_3, A_4, A_5\}$ scheme axiomatice

$$\left.\begin{array}{l} A_1:\ U \to (V \to U) \\ A_2:\ (U \to (V \to Z)) \longrightarrow ((U \to V) \longrightarrow (U \to Z)) \\ A_3:\ (U \to V) \to (\neg V \to \neg U) \end{array}\right\} \text{valabile și pt calculul propozițional}$$

$A_4:\ (\forall x)\, U(x) \to U(t)$, unde t este termen arbitrar

$A_5:\ (U \to V(y)) \longrightarrow (U \to (\forall x)\, V(x))$

y – variabilă liberă în V care nu apare în U

x – NU este var. liberă nici în U nici în V

- reguli de inferență

$\mathcal{R}_{Pr} = \{mp, gen\}$

modus ponens: $U, U \to V \vdash_{mp} V$

regula generalizării: $U(x) \vdash_{gen} (\forall x)\, U(x)$, x – var liberă în U

Definiții

variabile legate – sub incidența unui cuantificator

variabile libere – nu au cuantificator

formulă închisă – TOATE variabilele sunt legate

formulă deschisă – ear contrar ↺

Deducția

Definiția deducției

- Fie formulele $U_1, U_2, \ldots, U_n$ numite ipoteze și V formulă propozițională. Spunem că V este deductibilă din $U_1, U_2, \ldots, U_n$ și notăm $U_1, U_2, \ldots, U_{n-1}, U_n \vdash V$, dacă există o secvență de formule $(f_1, f_2, \ldots, f_m)$ a.î. $f_m = V$ și $\forall i \in \{1, \ldots, m\}$ avem:
  - $f_i \in \mathcal{A}_{Pr}$
  - $f_i \in \{U_1, U_2, \ldots, U_n\}$
  - $f_j, f_k \vdash_{mp} f_i$, $j < i$ și $k < i$
  - $f_j \vdash_{gen} f_i$, $j < i$

Secvența $(f_1, f_2, \dots, f_m)$ se numește deducția lui V din $U_1, U_2, \dots, U_m$.

Definiția teoremei ($\vdash$)

O formulă $U \in F_{Pn}$ a.î. $\emptyset \vdash U$ (sau $\vdash U$) se numește teoremă

Semantica logicii predicatelor

→ este furnizat un înțeles

Definiția interpretării

O interpretare pentru un limbaj $\mathcal{L}$ al calculului predicatelor este o pereche $i = \langle D, m \rangle$, unde:

- → D = mulțime nevidă numită domeniu al interpretării
- → m = funcție care asociază
  - o valoare fixă $m(c)$ din domeniul D unei constante c
  - o funcție $m(f) : D^n \to D$ fiecărui simbol de funcție f de aritate n
  - un predicat $m(P) : D^n \to \{T, F\}$ fiecărui simbol de predicat P de aritate n

Notații: pt $i = \langle D, m \rangle$

→ $|i| = D$ domeniul interpretării i

→ $i|x| = m(x)$, unde x – constantă / simbol de funcție / simbol de predicat

→ $As(i)$ = mulțimea funcțiilor de asignare de variabile peste domeniul interpretării i

O funcție $a \in As(i)$ este definită astfel: $a : Var \to |i|$

→ $[a]_x = \{a' \mid a' \in As(i)$ și $a'(y) = a(x)$ pt orice $y \neq x\}$

Definiția funcției de evaluare

Fie o interpretare i și $a \in As(i)$. Se definește inductiv funcția de evaluare v_a^i

- $v_a^i(x) = a(x)$, $x \in Var$
- $v_a^i(c) = i|c|$, $c \in Const$
- $v_a^i(f(t_1, \dots, t_m)) = i|f|(v_a^i(t_1), \dots, v_a^i(t_m))$, $f \in \mathcal{F}_k$, $m > 0$
- $v_a^i(P(t_1, \dots, t_m)) = i|P|(v_a^i(t_1), \dots, v_a^i(t_m))$, $P \in \mathcal{P}_k$, $m > 0$
- $v_a^i(\neg A) = \neg v_a^i(A)$; $v_a^i(A \vee B) = v_a^i(A) \vee v_a^i(B)$; $v_a^i(A \wedge B) = v_a^i(A) \wedge v_a^i(B)$
- $v_a^i(A \to B) = v_a^i(A) \to v_a^i(B)$
- $v_a^i((\exists x) A(x)) = T \iff v_{a'}^i(A(x)) = T$ pt o funcție $a' \in [a]_x$
- $v_a^i((\forall x) A(x)) = T \iff v_{a'}^i(A(x)) = T$ —‖—

Concepte semantice

O formulă A este:

→ realizabilă (consistentă) $\iff \exists$ o interpretare $\mathcal{I}$ și o funcție $a \in As(\mathcal{I})$ a.î. $v_a^{\mathcal{I}}(A) = T$

→ nerealizabilă (inconsistentă) → caz contrar

→ adevărată în interpretarea $\mathcal{I}$ $\iff$ pt $\forall$ funcție $a \in As(\mathcal{I})$ de asignare avem $v_a^{\mathcal{I}}(A) = T$ și notăm $\models_{\mathcal{I}} A$, iar $\mathcal{I}$ se numește model al lui A

→ model: $v_a^{\mathcal{I}}(A) = T$

→ antimodel: interpretarea $\mathcal{I}$ este evaluată ca falsă $v_a^{\mathcal{I}}(A) = F$

→ tautologie (validă) $\iff$ adev. în ORICE interpretare, notație: $\models A$

→ două formule A și B logic echivalente dacă:

$v_a^{\mathcal{I}}(A) = v_a^{\mathcal{I}}(B)$ pt orice interpretare $\mathcal{I}$ și funcție a de asignare

notație: $A \equiv B$

→ O mulțime S de formule implică logic o formulă A dacă toate modelele mulțimii (adică modelele conjuncției formulelor din S) sunt modele ale formulei. Spunem că A este o consecință logică a mulțimii de formule S și notăm $S \models A$

O mulțime de formule este:

→ consistentă: dacă formula obținută prin conjuncția elementelor sale este consistentă (adică are cel puțin un model)

→ inconsistentă: —— ‖ —— nu există nici un model pt formulă

! Obs:

- evaluarea unei formule A închise depinde doar de interpretarea în care se evaluează formula, notându-se $v^{\mathcal{I}}(A)$
- dacă interpretarea are domeniu finit:

→ o formulă cuantificată universal este înlocuită cu conjuncția instanțelor acesteia folosind toate elementele domeniului de interpretare.

→ o formulă cuantificată existențial este înlocuită cu disjuncția instanțelor acesteia folosind toate elementele domeniului de interpretare

Echivalențe logice

Legile de expansiune

$(\forall x)\, A(x) \equiv (\forall x)\, A(x) \wedge A(t)$

$(\exists x)\, A(x) \equiv (\exists x) A(x) \vee A(t)$, t - termen oarecare $t \neq x$

Legile infinite ale lui Morgan

$\neg (\exists x) A(x) \equiv (\forall x)\ \neg A(x)$

$\neg (\forall x)\, A(x) \equiv (\exists x) \neg A(x)$

Legile de interschimbare a cuantificatorilor

$(\exists x)\ (\exists y)\ A(x,y) \equiv (\exists y)(\exists x)\, A(x,y)$

$(\forall x)\ (\forall y) A(x,y) \equiv (\forall x)(\forall y)\ A(x,y)$

Legi de extragere

Legile de extragere a cuantificatorilor în fața formulei

$A \vee (\exists x) B(x) \equiv (\exists x)\ (A \vee B(x))$

$A \vee (\forall x)\ B(x) \equiv (\forall x)\ (A \vee B(x))$

$A \wedge (\exists x) B(x) \equiv (\exists x)(A \vee B(x))$, A nu conține pe x ca var. liberă sau legată

$A \wedge (\forall x) B(x) \equiv (\forall x)(A \vee B(x))$

(la fel și pt A în dreapta cuantificatorului)

Legile distributivității

- $\exists$ față de $\vee$: $(\exists x)(A(x) \vee B(x)) \equiv (\exists x)\ A(x) \vee (\exists x)\, B(x)$
- $\forall$ față de $\wedge$: $(\forall x)\ (A(x) \wedge B(x)) \equiv (\forall x)\ A(x) \wedge (\forall x)\ B(x)$

Semidistributivități

- $\exists$ față de $\wedge$: $\models (\exists x)\, (A(x) \wedge B(x)) \longrightarrow (\exists x)\ A(x) \wedge (\exists x)\, B(x)$
- $\forall$ față de $\vee$: $\models (\forall x)\ (A(x) \vee B(x)) \longrightarrow (\forall x)\, A(x) \vee (\exists x)\, B(x)$

Forme normale ale formulelor predicative

• O formulă U predicativă este în **forma normală prenexă** dacă este de forma $(Q_1 x_1)\dots(Q_m x_m)M$, unde $Q_i, i=\overline{1,m}$ sunt cuantificatori logici, iar M **nu** conține cuantificatori.

Secvența se numește "prefixul formulei", iar M este matricea formulei U.

• O formulă U predicativă este în **forma normală prenexă conjunctivă** dacă ea este în formă normală prenexă, iar matricea este în FNC($\wedge$).

Teoremă: Orice formulă din calculul predicatelor poate fi transformată într-o formă normală prenexă logic echivalentă.

Algoritmul de aducere la forma normală prenexă

Pas 1: înlocuirea $\rightarrow$ și $\leftrightarrow$ cu $\neg, \wedge, \vee$

Pas 2: Se aplică legile finite și infinite ale lui DeMorgan a.î. cuantificatorii să nu fie precedați de negație.

Pas 3: Se redenumesc variabilele legate a.î. ele să fie distincte.
sub incidența unui cuantif.

Pas 4: Se utilizează echivalențe logice care reprezintă legile de extragere a cuantificatorilor în fața formulei. ! Ordinea de extragere este arbitrară.

Pas 5: Forma normală Skolem

- fie U o formulă predicativă, iar $U^P = (Q_1 x_1)\dots(Q_m x_m)M$ una dintre formele sale normale prenexe

- Formulei U îi corespunde o formulă în forma normală Skolem notată U^S care se obține astfel: pt fiecare cuantif existențial $[(\exists)]Q_n$ din prefix se aplică:

→ dacă înaintea simbolului Q_n **nu** apare niciun cuantificator universal, atunci se alege o constantă notată a, diferită de toate constantele care apar în M și se înlocuiesc toate aparițiile variabilei x_n în M cu a. Se șterge $(Q_n x_n)$ din prefixul formulei.

→ dacă înaintea simbolului Q_n apar cuantif. universali $Q_{s_1}, \dots Q_{s_m}$. $(\forall)$ unde $1 \le s_1 < \dots < s_m < n$, atunci alegem un simbol f în funcție de m variabile, diferit de celelalte simboluri de funcții și se înlocuiește fiecare apariție a variabilei în M cu $f(x_{s_1}, \dots, x_{s_m})$. Se șterge $(Q_n x_n)$ din prefixul formulei.

constante Skolem } → folosite pt a înlocui variabilele existențiale
funcții Skolem }

exemplu $(\exists x)(\exists y)(\forall z)(\exists t)(\forall s)(\exists u)(\forall w)\, P(x, y, z, t, s, u, w)$

înlocuim cu const pt că nu apar $\forall$ înainte

înlocuim cu funcții ~~cu var~~

$x \leftarrow a$

$y \leftarrow b$

$t \leftarrow f(z)$

$u \leftarrow g(z, s)$

Forma Skolem: $(\forall z)(\forall s)(\forall w)\, P(a, b, z, f(z), s, g(z,s), w)$

Forma normală clauzală

Pas 6: Formulei U îi corespunde o formulă în forma normală skolem fără cuantif. notată U^{S2} care se obține prin eliminarea cuantificatorilor universali din U^S

Pas 7: Formulei U îi corespunde o formulă în forma normală clauzală notată U^C se obține din U^{S2} prin aducerea la FNC.

!Obs

Transformările utilizate în procesul de Skolemizare nu păstrează echivalența logică, dar păstrează inconsistența.

Teoreme:

Fie $U_1, U_2, \ldots, U_n, V$ formule predicative.

- V inconsistentă $\iff$ V^P inconsistentă $\iff$ V^S inconsistentă $\iff$ V^{S2} inconsistentă $\iff$ V^C inconsistentă
- $\{U_1, U_2, \ldots, U_n\}$ inconsistentă $\iff$ $\{U_1^C, U_2^C, \ldots, U_n^C\}$ inconsistentă

Sumar

Pas 5: eliminarea cuantificatorilor $\exists$ (Forma normală Skolem)

Pas 6: eliminarea cuantificatorilor $\forall$ (Forma normală Skolem fără cuantificatori)

Pas 7: aducerea la Forma Normală Clauzală (distributivitatea $\vee$ față de $\wedge$)

Teorema de completitudine și corectitudine

Fie S o mulțime de formule predicative, iar A o formulă predicativă.

→ completitudinea: dacă $S \models A$ atunci $S \vdash A$

→ corectitudinea: dacă $S \vdash A$ atunci $S \models A$

Teorema respingerii

Fie S o mulțime de formule predicative, iar A o formulă predicativă

Dacă $S \cup \{\neg A\}$ este inconsistentă atunci $S \vdash A$.

Teorema de deducție

Fie S o mulțime de formule predicative, iar A o formulă predicativă.
Dacă $S \cup \{A\} \vdash B$ atunci $S \vdash A \to B$

Teorema lui Church, 1936

- Problema validității unei formule în calculul predicatelor de ordin I este nedecidabilă. Mulțimea formulelor valide din acest sistem logic este recursiv numărabilă, adică există o procedură ? care, având ca intrare o formulă A din limbaj, are următorul comportament:

→ dacă formula A este validă, ? se termină și furnizează răspunsul corespunzător
→ dacă formula A nu este validă, ? se termină cu răspunsul corespunzător SAU execuția procedurii nu se încheie niciodată.

- Calculul predicatelor este semidecidabil.

Curs 9

Metoda tabelelor semantice

—————————||—————————— (pag 9)

Clase de formule

clasa α – formule de tip conjunctiv

$A \wedge B$ → A → B

$\neg(A \vee B)$ → $\neg A$ → $\neg B$

$\neg(A \rightarrow B)$ → A → $\neg B$

clasa β – formule de tip disjunctiv

$A \vee B$ → A | B

$\neg(A \wedge B)$ → $\neg A$ | $\neg B$

$A \rightarrow B$ → $\neg A$ | B

clasa γ (gamma) – formule cuantificate universal

$(\forall x)\ A(x)$

$\neg(\exists x)\ A(x)$

$(\forall x) A(x)$ → $A(c_1)$ → $A(c_2)$ → … → $A(c_m)$ → $(\forall x) A(x)$ copie formulă

$c_1, \ldots, c_m$ toate formulele existente pe ramură

$\neg(\exists x) A(x)$ → $\neg A(c_1)$ → $\neg A(c_2)$ → … → $\neg A(c_m)$ → $\neg(\exists x) A(x)$ copie formulă

clasa δ (delta) – formule cuantificate existențial

$(\exists x)\ A(x)$

$\neg(\forall x)\ A(x)$

$(\exists x) A(x)$ → $A(a)$

$\neg(\forall x) A(x)$ → $\neg A(a)$

a – constantă nou introdusă

Arborele binar de descompunere a unei formule

—————————||——————————

→ extinderea se încheie:
- dacă pe ramură apare o formulă și negația sa
- dacă s-au descompus toate formulele sau prin aplicarea regulilor de descompunere nu se mai obțin formule noi pe acea ramură

Obs

—||—

– se recomandă utilizarea regulilor de tip δ (se introduc constante noi) înaintea regulilor γ (se folosesc cele existente)

Semi-decidabilitatea calculului predicativ

- pt cazul logicii predicatelor de ordinul I, arborele poate fi infinit datorită combinării regulilor de tip γ și δ.
- dacă arborele asociat negației unei formule predicative este finit, atunci se poate decide dacă formula respectivă este o tautologie sau nu, dar dacă arborele este infinit, nu se poate decide nimic asupra validității formulei

Substituții

Def: O substituție este o funcție definită pe mulțimea variabilelor, Var cu valori în mulțimea termenilor, TERM. Se notează cu $\theta = [x_1 \leftarrow t_1, \ldots, x_k \leftarrow t_k]$ reprezentând o mulțime finită de înlocuiri de variabile cu termeni.
$x_1, \ldots, x_k$ sunt variabile distincte
$t_1, \ldots, t_k$ sunt termeni, a.î. $\forall i = \overline{1,k}$, $t_i \neq x_i$ și x_i nu este subterment al lui t_i

- $dom(\theta) = \{x_1, \ldots, x_k\}$ → domeniul substituției θ
- ε - substituția vidă (epsilon)
- $\varphi, \varepsilon, \psi, \eta, \theta, \lambda$

Aplicarea substituției

$\theta = [x_1 \leftarrow t_1, \ldots, x_k \leftarrow t_k]$ asupra formului U se definește recursiv

- $\theta(x_i) = t_i$, $x_i \in dom(\theta)$; $\theta(x) = x$, $x \notin dom(\theta)$
- $\theta(c) = c$, c - constantă
- $\theta(f(t_1, \ldots, t_m)) = f(\theta(t_1), \ldots, \theta(t_m))$, $f \in \mathcal{F}_m$
- $\theta(P(t_1, \ldots, t_m)) = P(\theta(t_1), \ldots, \theta(t_m))$, $P \in \mathcal{P}_m$
- $\theta(\neg U) = \neg \theta(U)$
- $\theta(U \wedge V) = \theta(U) \wedge \theta(V)$
- $\theta(U \vee V) = \theta(U) \vee \theta(V)$
- $\theta(U \rightarrow V) = \theta(U) \rightarrow \theta(V)$
- $\theta(U \leftrightarrow V) = \theta(U) \leftrightarrow \theta(V)$

Compunerea substituțiilor

$\theta_1 = [x_1 \leftarrow t_1, \ldots, x_k \leftarrow t_k]$

$\theta_2 = [y_1 \leftarrow s_1, \ldots, y_k \leftarrow s_k]$

$\theta = \theta_1 \circ \theta_2 = [x_i \leftarrow \theta_2(t_i) \mid x_i \in dom(\theta_1), x_i \neq \theta_2(t_i)] \cup [y_j \leftarrow s_j \mid y_j \in dom(\theta_2) \setminus dom(\theta_1)]$

! Obs
Nu întotdeauna compunerea unor substituții este o substituție

Proprietăți ale operației de compunere

→ element neutru: ε – substituția vidă:

$\varepsilon\theta = \theta\varepsilon = \theta$, $\forall\, \theta$ o substituție

→ asociativitatea

$\theta_1(\theta_2\theta_3) = (\theta_1\theta_2)\theta_3 = \theta_1\theta_2\theta_3$

→ în general compunerea nu este comutativă

Unificatori

O substituție θ se numește unificator al termenilor t_1 și t_2 dacă $\theta(t_1) = \theta(t_2)$. Termenul $\theta(t_1)$ se numește instanța comună a termenilor unificați.

Un unificator al mulțimii de formule $\{U_1, U_2, \dots, U_m\}$ este o substituție θ cu proprietatea $\theta(U_1) = \dots = \theta(U_m)$

Cel mai general unificator (mgu) este unificator μ cu proprietatea că orice alt unificator θ se obține din compunerea lui μ cu o altă subst. λ:

$$\theta = \mu\lambda$$

Algoritm pentru determinarea celui mai general unificator a doi literali

Date de intrare: $l_1 = P_1(t_{1_1}, \dots, t_{1_m})$ și $l_2 = P_2(t_{2_1}, \dots, t_{2_k})$ doi literali

Date de ieșire: $mgu(l_1, l_2)$ sau „l_1, l_2 nu sunt unificabili"

```
dacă P1 ≠ P2 //simbolurile predicative sunt diferite
   atunci scrie „l1, l2 nu sunt unificabili"; STOP
□
dacă n ≠ k //aritate diferită pt acelaşi simbol predicativ
   atunci scrie „l1, l2 nu sunt unificabili"; STOP
□
θ ← ε //inițializare cu substituția vidă
cât timp (θ(l1) ≠ θ(l2))
   Din θ(l1), θ(l2) se determină cele mai din stânga simbol de funcție, constantă
   sau variabilă diferite și notăm cu t1, și t2 termenii lor corespunzători
   dacă (niciunul dintre t1 și t2 nu este variabilă sau unul este subtermenul celuilalt)
      atunci scrie „l1, l2 nu sunt unificabili"; STOP
   □
   dacă (t1 este variabila)
      atunci λ = [t1 ← t2]
      altfel λ = [t2 ← t1]
   □
   θ ← θλ
   dacă (θ nu este substituție)
      atunci scrie „l1, l2 nu sunt unificabili"; STOP
   □
□
scrie „l1 și l2 sunt unificabili, mgu(l1, l2) =" θ
```

Curs 10

Sistem formal (axiomatic) asociat Rezoluției predicative

$Res^{Pr} = (\Sigma^{Pr}_{Res}, \mathcal{F}^{Pr}_{Res}, A^{Pr}_{Res}, R^{Pr}_{Res})$

- alfabetul

$\Sigma^{Pr}_{Res} = \Sigma_{Pr} \setminus \{\forall, \exists, \wedge, \rightarrow, \leftrightarrow\}$

- mulțimea formulelor bine-definite

$\mathcal{F}^{Pr}_{Res} \cup \{\square\}$

$\mathcal{F}^{Pr}_{Res}$ – mulțimea tuturor clauzelor ce se pot forma cu Σ^{Pr}_{Res}

$\square$ – clauza vidă

- mulțimea axiomelor

$A^{Pr}_{Res} = \emptyset$

- mulțimea regulilor de inferență

$R^{Pr}_{Res} = \{res^{Pr}, fact\}$

• regula rezoluției predicative

$A \vee l_1, B \vee \neg l_2 \vdash^{Pr}_{res} \theta(A) \vee \theta(B)$, unde $\theta = mgu(l_1, l_2)$ și $A, B \in \mathcal{F}^{Pr}_{Res}$

→ $C_1 = A \vee l_1$, $C_2 = B \vee \neg l_2$ clauzele care rezolvă, dacă literalii l_1 și l_2 sunt unificabili

→ rezolventul binar $C_3 = Res^{Pr}_{\theta}(C_1, C_2) = \theta(A) \vee \theta(B)$

• regula factorizării

$C \vdash_{fact} C'$, C' – factor al lui C, unde $C = l_1 \vee l_2 \vee \dots \vee l_k \vee l_{k+1} \vee \dots \vee l_m$,

$\lambda = mgu(l_1, l_2, \dots, l_k)$

$C' = \lambda(l_k) \vee \lambda(l_{k+1}) \vee \dots \vee \lambda(l_m)$

exemple curs

Teoremă

Fie $U_1, U_2, \dots, U_n$ și V formule predicative.

• $\vdash V \iff (\neg V)^c \vdash^{Pr}_{res} \square$

• $U_1, U_2, \dots, U_n \vdash V \iff \{U_1^c, U_2^c, \dots, U_n^c, (\neg V)^c\} \vdash_{res} \square$

Obs

Variabilele din clauze distincte se recomandă să fie distincte.

Algoritmul rezoluției predicative

Date de intrare $U_1, U_2, \ldots, U_n, V$ - formule predicative

Date de ieșire: „au loc $U_1, U_2, \ldots, U_n \vdash V$" **sau** „nu au loc $U_1, U_2, \ldots, U_n \vdash V$"

se construiește $S = \{U_1^c, U_2^c, \ldots, U_n^c, (\neg V)^c\}$

Repetă

Se selectează literalii l_1, l_2 și clauzele C_1, C_2 a.î. sunt clauze sau factori ai unor clauze din S

Fie $l_1 \in C_1$ și $\neg l_2 \in C_2$, respectiv $l_1, l_2 \in C_1$

Dacă l_1 și l_2 sunt unificabili cu $\theta = mgu(l_1, l_2)$

Atunci

$C = Res_\theta^{l_1 l_2}(C_1, C_2)$ respectiv $C = fact_\theta(C_1)$

Dacă $C = \square$

Atunci scrie „au loc $U_1, U_2, \ldots, U_n \vdash V$"; **STOP**

Altfel $S = S \cup \{C\}$

$\square$

$\square$

până când nu și mai pot deriva noi rezolvenți **sau** un număr fixat de iterații au fost executate

dacă nu se pot deriva noi rezolvenți

atunci scrie „nu au loc $U_1, U_2, \ldots, U_n \vdash V$"

$\square$ **altfel** scrie „nu se poate decide dacă au loc sau nu $U_1, U_2, \ldots, U_n \vdash V$"

Strategii și rafinări

—————— // ——————

Completitudinea și corectitudinea

—— // ——

Completitudinea rezoluției de intrare

Def: clauză
- pozitivă dacă conține literali pozitivi
- negativă dacă conține literali negativi

clauză Horn = conține un singur literal pozitiv, ceilalți fiind negativi

Teoremă:

Rezoluția de intrare este completă pe o mulțime de clauze Horn, cu o clauză negativă ca și clauză de vârf (prolog)

$H_i : l_1 \wedge l_2 \wedge \ldots \wedge l_m \rightarrow r \quad , i \in \{1, \ldots k\}$

$C : m_1 \wedge m_2 \wedge \ldots \wedge m_m$?