Fundamentele limbajelor de programare
CURS 6

Andrei Sipoș[1]

Facultatea de Matematică și Informatică, DL Info, Anul II
Semestrul II, 2025/2026

[1]Curs realizat împreună cu Traian Florin Șerbănuță.

Clauze

Dacă avem o formulă φ, $n \in \mathbb{N}$ și $k_1 < \ldots < k_n$ cu $FV(\varphi) = \{x_{k_1}, \ldots, x_{k_n}\}$, notăm cu $\forall\varphi$ enunțul $\forall x_{k_1} \ldots \forall x_{k_n} \varphi$.

Numim **literal** o formulă de forma φ sau $\neg\varphi$, unde φ este o formulă atomică relațională (literalul este, respectiv, **pozitiv** sau **negativ**). Numim **clauză** o formulă de forma $\forall(L_1 \vee \ldots \vee L_m)$, unde $L_1, \ldots, L_m$ sunt literali. Numim **clauză definită** o clauză unde apare exact un literal pozitiv, anume pe prima poziție. Dacă $A_0, \ldots A_m$ sunt formule atomice relaționale, atunci clauza

$$\forall(A_0 \vee \neg A_1 \vee \ldots \vee \neg A_m)$$

se scrie și sub forma (cunoscută din limbajul Prolog)

$$A_0 \leftarrow A_1, \ldots, A_m.$$

Un **program** va fi o mulțime finită de clauze definite. Un **scop definit** este o clauză în care apar doar literali negativi.

Structuri Herbrand

De acum încolo, vom face presupunerea că există măcar o constantă în signatură, așadar $\widetilde{T}_\sigma \neq \emptyset$. Notăm cu B_σ (numită **baza Herbrand**) mulțimea tuturor σ-formulelor atomice relaționale fără variabile.

Spunem că o σ-structură este **Herbrand** atunci când universul ei este $\widetilde{T}_\sigma$, iar simbolurile de funcție sunt interpretate „de ele însele". Observăm că o σ-structură Herbrand este complet determinată de mulțimea J a acelor formule din B_σ adevărate în ea. Pentru orice submulțime J a lui B_σ și orice enunț φ, spunem că $J \models_H \varphi$ atunci când structura Herbrand asociată lui J satisface φ.

Dacă $\mathcal{A}$ este o σ-structură, vom nota $J_\mathcal{A} := \{\varphi \in B_\sigma \mid \mathcal{A} \models \varphi\}$.

Proprietăți

Teoremă

Fie $\mathcal{A}$ o σ-structură și φ o clauză definită (sau un scop definit) cu $\mathcal{A} \models \varphi$. Atunci $J_{\mathcal{A}} \models_H \varphi$.

Demonstrație

Presupunem φ clauză definită. Avem că există $m, n \in \mathbb{N}$, formule atomice relaționale $A_0, A_1, \ldots, A_m$ și variabile $x_1, \ldots, x_n$ cu

$$\varphi = \forall x_1 \ldots \forall x_n (A_0 \vee \neg A_1 \vee \ldots \vee \neg A_m).$$

Fie $t_1, \ldots, t_n \in \widetilde{T}_\sigma$. Notăm, pentru orice i, $A_i' := A_i[x_1 := t_1] \ldots [x_n := t_n]$ și $\varphi' := A_0' \vee \neg A_1' \vee \ldots \vee \neg A_m'$. Vrem $J_{\mathcal{A}} \models_H \varphi'$. Presupunem că, pentru orice $i \geq 1$, $J_{\mathcal{A}} \models_H A_i'$. Arătăm că $J_{\mathcal{A}} \models_H A_0'$. Deci, pentru orice $i \geq 1$, cum $A_i' \in B_\sigma$, $A_i' \in J_{\mathcal{A}}$, deci $\mathcal{A} \models A_i'$. Cum $\mathcal{A} \models \varphi$, avem $\mathcal{A} \models \varphi'$, iar, cum, pentru orice $i \geq 1$, $\mathcal{A} \models A_i'$, avem $\mathcal{A} \models A_0'$. Cum $A_0' \in B_\sigma$, $A_0' \in J_{\mathcal{A}}$, deci $J_{\mathcal{A}} \models_H A_0'$.

Proprietăți

Teoremă

Fie P un program. Atunci $K_P := \{J \subseteq B_\sigma \mid J \models_H P\}$ este o mulțime Moore pe B_σ.

Demonstrație

Faptul că $B_\sigma \models_H P$ rămâne ca exercițiu. Fie acum $K \subseteq K_P$ cu $K \neq \emptyset$. Vrem $\bigcap K \models_H P$. Fie $\varphi \in P$. Fie $t_1, \ldots, t_n \in \widetilde{T}_\sigma$. Folosim aceleași notații ca în demonstrația precedentă. Presupunem că, pentru orice $i \geq 1$, $\bigcap K \models_H A'_i$. Arătăm că $\bigcap K \models_H A'_0$. Deci, pentru orice $i \geq 1$, cum $A'_i \in B_\sigma$, $A'_i \in \bigcap K$, deci, pentru orice $J \in K$, $A'_i \in J$, adică $J \models_H A'_i$. Fie $J \in K$. Avem $J \models_H \varphi$, deci $J \models_H \varphi'$, iar, cum, pentru orice $i \geq 1$, $J \models_H A'_i$, avem $J \models_H A'_0$, deci, cum $A'_0 \in B_\sigma$, $A'_0 \in J$. Așadar, $A'_0 \in \bigcap K$, deci $\bigcap K \models_H A'_0$.

Pentru orice program P, notăm $M_P := \bigcap K_P \in K_P$.

Proprietăți

Teoremă

Pentru orice program P, $M_P = \{\varphi \in B_\sigma \mid P \models \varphi\}$.

Demonstrație

Pentru „$\supseteq$”, fie $\varphi \in B_\sigma$ cu $P \models \varphi$. Fie $J \in K_P$. Vrem $\varphi \in J$. Cum $J \models_H P$, avem $J \models_H \varphi$, iar, cum $\varphi \in B_\sigma$, $\varphi \in J$.

Pentru „$\subseteq$”, fie $\varphi \in M_P$. Fie $\mathcal{A}$ cu $\mathcal{A} \models P$. Vrem $\mathcal{A} \models \varphi$. Cum P este o mulțime de clauze definite, dintr-o teoremă anterioară avem $J_\mathcal{A} \models_H P$, deci $J_\mathcal{A} \in K_P$. Rezultă $M_P \subseteq J_\mathcal{A}$, deci $\varphi \in J_\mathcal{A}$, adică $J_\mathcal{A} \models_H \varphi$. Rezultă $\mathcal{A} \models \varphi$.

Clauze și scopuri

Reamintim că o **clauză (definită)** era o formulă de forma

$$\forall(A_0 \vee \neg A_1 \vee \ldots \vee \neg A_m),$$

unde $A_0, \ldots, A_m$ erau formule atomice relaționale, un **program** era o mulțime finită de clauze, iar un **scop** era o formulă de forma

$$\forall(\neg A_1 \vee \ldots \vee \neg A_m).$$

Formula de mai sus este echivalentă semantic cu

$$\neg\exists(A_1 \wedge \ldots \wedge A_m),$$

care corespunde intuitiv ideii de a căuta o soluție.

Pentru orice formulă fără cuantificatori φ și orice substituție θ, notăm cu $\varphi\theta$ formula care se obține din φ aplicând pe θ pe toate variabilele sale libere.

Regula rezoluției

Fie G, G' scopuri și C clauză cu $Var(G) \cap Var(C) = \emptyset$. Fie m, $k \in \mathbb{N}$ astfel încât $G = \forall(\neg A_1 \vee \ldots \vee \neg A_m)$ și $C = \forall(B_0 \vee \neg B_1 \vee \ldots \vee \neg B_k)$. Considerăm B_0 ca fiind de forma $p(t_1, \ldots, t_n)$. Fie $i \leq m$ astfel încât A_i este de forma $p(s_1, \ldots, s_n)$. Fie θ un cgu al lui A_i și B_0, adică al mulțimii $\{s_1 = t_1, \ldots, s_n = t_n\}$. Spunem că G' este **derivat prin rezoluție** din G, C și θ, și notăm $(G, C, \theta) \triangleright G'$, dacă

$$G' = \forall(\neg A_1 \vee \ldots \vee \neg A_{i-1} \vee \neg B_1 \vee \ldots \vee \neg B_k \vee \neg A_{i+1} \vee \ldots \vee \neg A_m)\theta.$$

De acum încolo, vom fixa P un program.

Derivări

Fie $a \in \mathbb{N}^* \cup \{\mathbb{N}\}$. Numim o P-**derivare (prin rezoluție)** un triplet $((G_i)_{i<a}, (C_i)_{i+1<a}, (\theta_i)_{i+1<a})$, astfel încât, pentru orice i cu $i+1 < a$, C_i este o clauză obținută dintr-o clauză din P prin redenumirea variabilelor sale și $(G_i, C_i, \theta_i) \triangleright G_{i+1}$.

Fie $n \in \mathbb{N}^*$ și $((G_i)_{i<n}, (C_i)_{i+1<n}, (\theta_i)_{i+1<n})$ o P-derivare. Spunem că **substituția calculată** a sa este $\widetilde{\theta}_{n-2} \circ \ldots \circ \widetilde{\theta}_1 \circ \theta_0$.

Fie G un scop, $n \in \mathbb{N}^*$ și $((G_i)_{i<n}, (C_i)_{i+1<n}, (\theta_i)_{i+1<n})$ o P-derivare cu $G_0 = G$ și $G_{n-1} = \perp$. Atunci spunem că derivarea este o P-**respingere** a lui G, iar substituția sa calculată spunem că este o P-**soluție** a lui G.

Fie G un scop, $n \in \mathbb{N}^*$ și $((G_i)_{i<n}, (C_i)_{i+1<n}, (\theta_i)_{i+1<n})$ o P-derivare cu $G_0 = G$ și $G_{n-1} \neq \perp$ care nu admite o prelungire la una de lungime $n+1$. Atunci spunem că derivarea este o P-**derivare eșuată** a lui G.

Teorema de corectitudine

Fie $m \in \mathbb{N}^*$, $A_1, \ldots, A_m$ formule atomice relaționale și θ o P-soluție a lui $\forall(\neg A_1 \vee \ldots \vee \neg A_m)$. Atunci $P \models \forall(A_1 \wedge \ldots \wedge A_m)\theta$.

Demonstrație

Demonstrăm după lungimea P-respingerii. Pasul de bază are loc atunci când avem o singură aplicare a rezoluției, așadar trebuie să avem $m = 1$, iar clauza folosită are tot lungime 1, fie ea $\forall B_0$. Atunci θ este cgu pentru A_1 și B_0, deci $A_1\theta = B_0\theta$. Cum $\forall B_0$ este o redenumire a unei clauze din P, avem $P \models \forall B_0$, deci $P \models \forall B_0\theta$, așadar $P \models \forall A_1\theta$, ceea ce trebuia demonstrat.

Teorema de corectitudine

Demonstrație (cont.)

Pentru pasul de inducție, notăm cu $((G_i)_{i<n}, (C_i)_{i+1<n}, (\theta_i)_{i+1<n})$ P-respingerea. Luăm C_0 de forma $\forall(B_0 \vee \neg B_1 \vee \ldots \vee \neg B_k)$ și fie i astfel încât θ_0 este cgu al lui A_i și B_0. Așadar, G_1 este

$$\forall(\neg A_1 \vee \ldots \vee \neg A_{i-1} \vee \neg B_1 \vee \ldots \vee \neg B_k \vee \neg A_{i+1} \vee \ldots \vee \neg A_m)\theta_0.$$

Din ipoteza de inducție, avem

$$P \models \forall(A_1 \wedge \ldots \wedge A_{i-1} \wedge B_1 \wedge \ldots \wedge B_k \wedge A_{i+1} \wedge \ldots \wedge A_m)\theta.$$

Cum C_0 este o redenumire a unei clauze din P, avem

$$P \models \forall(B_0 \vee \neg B_1 \vee \ldots \vee \neg B_k)\theta,$$

deci $P \models \forall(A_1 \wedge \ldots \wedge A_{i-1} \wedge B_0 \wedge A_{i+1} \wedge \ldots \wedge A_m)\theta$. Cum $B_0\theta_0 = A_i\theta_0$, $B_0\theta = A_i\theta$, deci $P \models \forall(A_1 \wedge \ldots \wedge A_m)\theta$.

O propoziție

Propoziție

Fie G un scop astfel încât există o P-respingere a lui G. Atunci $P \cup \{G\}$ este nesatisfiabilă.

Demonstrație

Scriem $G = \forall(\neg A_1 \vee \ldots \vee \neg A_m)$. Din teorema de corectitudine, rezultă $P \models \exists(A_1 \wedge \ldots \wedge A_m)$, deci

$$P \models \neg\forall(\neg A_1 \vee \ldots \vee \neg A_m) = \neg G,$$

de unde obținem concluzia.

Operatorul T_P

Putem defini operatorul $T_P : \mathcal{P}(B_\sigma) \to \mathcal{P}(B_\sigma)$ în felul următor: pentru orice J, $T_P(J)$ este mulțimea acelor $\varphi \in B_\sigma$ cu proprietatea că există $A_1, \ldots, A_m \in J$ astfel încât $\varphi \vee \neg A_1 \vee \ldots \vee \neg A_m$ este instanță a unei clauze din P.

Propoziție

Pentru orice J, $J \models_H P$ dacă și numai dacă $T_P(J) \subseteq J$.

Corolar

Pentru orice φ, $\varphi \in M_P$ dacă și numai dacă, pentru orice J cu $T_P(J) \subseteq J$, $\varphi \in J$.

Se observă și că T_P este monoton și, deci, din cele de mai sus, $M_P = \mu T_P$.

Mulțimea S_P

Definim mulțimea S_P ca fiind submulțimea lui B_σ a acelor φ cu proprietatea că există o P-respingere a lui $\neg\varphi$.

Propoziție

$S_P \subseteq M_P$.

Demonstrație

Fie $\varphi \in S_P$, adică există o P-respingere a lui $\neg\varphi$. Din Teorema de corectitudine, rezultă $P \models \varphi$, adică $\varphi \in M_P$.

Spre completitudine

Propoziție

$M_P \subseteq S_P$.

Demonstrație

Din cele spuse mai devreme, este suficient să arătăm că $T_P(S_P) \subseteq S_P$. Fie $\varphi \in T_P(S_P)$. Atunci $\varphi \in B_\sigma$ și există $A_1, \ldots, A_m \in S_P$ astfel încât $\varphi \vee \neg A_1 \vee \ldots \vee \neg A_m$ este instanță a unei clauze din P. Avem că există P-respingeri pentru $\neg A_1, \ldots, \neg A_m$, iar, punându-le cap la cap (nu detaliem cum, dar este important că sunt toate din B_σ), obținem o P-respingere pentru $\neg\varphi$, deci $\varphi \in S_P$.

Corolar

Pentru orice $\varphi \in B_P$ cu $P \models \varphi$, $\varphi \in S_P$.

Spre completitudine

Lemă

Fie φ o formulă atomică relațională cu $P \models \forall\varphi$. Atunci există o P-respingere a lui $\neg\varphi$ cu substituția calculată fiind identitatea.

Demonstrație

Scriem $Var(\varphi) = \{z_1, \ldots, z_n\}$. Introducem constante noi $a_1, \ldots, a_n$ și considerăm substituția θ care duce, pentru orice i, z_i în a_i. Atunci $\varphi\theta \in B_\sigma$, deci, din corolarul anterior, există o P-respingere a lui $\neg\varphi\theta$. Clar, dat fiind că nu apar variabile, substituția calculată trebuie să fie identitatea. Schimbăm în substituție din nou a_i-urile cu x_i-uri (de ce putem face asta?) și obținem P-respingerea cerută.

Teorema de completitudine

Teorema de completitudine

Fie $m \in \mathbb{N}^*$, $A_1, \ldots, A_m$ formule atomice relaționale și σ o substituție astfel încât $P \models \forall(A_1 \wedge \ldots \wedge A_m)\sigma$. Atunci există o P-soluție θ pentru $\forall(\neg A_1 \vee \ldots \vee \neg A_m)$ astfel încât $\forall(A_1 \wedge \ldots \wedge A_m)\sigma$ este o instanță a lui $\forall(A_1 \wedge \ldots \wedge A_m)\theta$.

Demonstrație (schiță)

Pentru orice i, avem $P \models \forall A_i\sigma$, deci, din lemă, există o P-respingere a lui $\neg A_i\sigma$ cu substituția calculată fiind identitatea. Din nou, putem pune cap la cap și obțin o P-respingere a lui $\forall(\neg A_1 \vee \ldots \vee \neg A_m)\sigma$ cu substituția calculată fiind identitatea. Aplicând un rezultat tehnic (Lema de ridicare), putem obține o P-respingere a lui $\forall(\neg A_1 \vee \ldots \vee \neg A_m)$, din ale cărei informații obținem substituția cerută.