Programare Logică – Seminariile V, VI, VII

Claudia MUREŞAN
Universitatea din Bucureşti, Facultatea de Matematică şi Informatică
c.muresan@yahoo.com, cmuresan@fmi.unibuc.ro

2019–2020, Semestrul II

Exerciţiul 1. Considerăm un limbaj de ordinul I conţinând un simbol de operaţie unară f, un simbol de constantă a şi simbol de relaţie binară r. Considerăm şi patru variabile distincte v, x, y şi z. Să se pună următorul enunţ într–o formă Skolem şi să se aplice algoritmul Davis–Putnam acelei forme Skolem:

$$\varepsilon = \exists\, v\, \forall\, x\, [[\exists\, y\, r(v, f(y)) \leftrightarrow \forall\, z\, r(f(f(z)), f(a))] \rightarrow r(a, f(x))].$$

Rezolvare:

$$\varepsilon = \exists\, v\, \forall\, x\, [[\exists\, y\, r(v, f(y)) \leftrightarrow \forall\, z\, r(f(f(z)), f(a))] \rightarrow r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\neg\, [\exists\, y\, r(v, f(y)) \leftrightarrow \forall\, z\, r(f(f(z)), f(a))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\neg\, [[\exists\, y\, r(v, f(y)) \rightarrow \forall\, z\, r(f(f(z)), f(a))] \wedge [\forall\, z\, r(f(f(z)), f(a)) \rightarrow \exists\, y\, r(v, f(y))]] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\neg\, [[\neg\, \exists\, y\, r(v, f(y)) \vee \forall\, z\, r(f(f(z)), f(a))] \wedge [\neg\, \forall\, z\, r(f(f(z)), f(a)) \vee \exists\, y\, r(v, f(y))]] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\neg\, [[\forall\, y\, \neg\, r(v, f(y)) \vee \forall\, z\, r(f(f(z)), f(a))] \wedge [\exists\, z\, \neg\, r(f(f(z)), f(a)) \vee \exists\, y\, r(v, f(y))]] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\neg\, [\forall\, y\, \neg\, r(v, f(y)) \vee \forall\, z\, r(f(f(z)), f(a))] \vee \neg\, [\exists\, z\, \neg\, r(f(f(z)), f(a)) \vee \exists\, y\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [[\neg\, \forall\, y\, \neg\, r(v, f(y)) \wedge \neg\, \forall\, z\, r(f(f(z)), f(a))] \vee [\neg\, \exists\, z\, \neg\, r(f(f(z)), f(a)) \wedge \neg\, \exists\, y\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [[\exists\, y\, \neg\, \neg\, r(v, f(y)) \wedge \exists\, z\, \neg\, r(f(f(z)), f(a))] \vee [\forall\, z\, \neg\, \neg\, r(f(f(z)), f(a)) \wedge \forall\, y\, \neg\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [[\exists\, y\, r(v, f(y)) \wedge \exists\, z\, \neg\, r(f(f(z)), f(a))] \vee [\forall\, z\, r(f(f(z)), f(a)) \wedge \forall\, y\, \neg\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\exists\, y\, [r(v, f(y)) \wedge \exists\, z\, \neg\, r(f(f(z)), f(a))] \vee \forall\, z\, [r(f(f(z)), f(a)) \wedge \forall\, y\, \neg\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\exists\, y\, \exists\, z\, [r(v, f(y)) \wedge \neg\, r(f(f(z)), f(a))] \vee \forall\, z\, \forall\, y\, [r(f(f(z)), f(a)) \wedge \neg\, r(v, f(y))] \vee r(a, f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\exists\, y\, \exists\, z\, [r(v, f(y)) \wedge \neg\, r(f(f(z)), f(a))] \vee \forall\, y\, \forall\, z\, [r(f(f(z)), f(a)) \wedge \neg\, r(v, f(y))] \vee r(a, f(x))].$$

Redenumim variabilele, astfel încât nicio variabilă să nu apară cuantificată şi existenţial, şi universal. Dacă am avea o formulă arbitrară, nu neapărat un enunţ, atunci ar trebui să redenumim variabilele şi pentru a nu avea variabile care să apară şi libere, şi legate.

Ca o paranteză, putem observa că avem posibilitatea de a efectua această redenumire a variabilelor calculând valoarea de adevăr a enunţului ε într–o algebră $\mathcal{A}$ cu mulţimea elementelor A, înzestrată cu o operaţie unară $f^{\mathcal{A}} : A \rightarrow A$, o constantă $a^{\mathcal{A}} \in A$ şi o relaţie binară $r^{\mathcal{A}} \subseteq A^2 = A \times A$. Amintesc că, întrucât ε este un enunţ, pentru orice interpretare $s : Var \rightarrow A$, valoarea de adevăr a enunţului ε în algebra $\mathcal{A}$ este $||\varepsilon||_{\mathcal{A}} = s(\varepsilon)$: nu depinde de interpretarea s, ci doar de algebra $\mathcal{A}$. Fie, aşadar, $s : Var \rightarrow A$ o interpretare arbitrară; avem:

$$||\varepsilon||_{\mathcal{A}} = ||\exists\, v\, \forall\, x\, [\exists\, y\, \exists\, z\, [r(v, f(y)) \wedge \neg\, r(f(f(z)), f(a))] \vee \forall\, y\, \forall\, z\, [r(f(f(z)), f(a)) \wedge \neg\, r(v, f(y))] \vee r(a, f(x))]||_{\mathcal{A}} =$$

$$s(\exists\, v\, \forall\, x\, [\exists\, y\, \exists\, z\, [r(v, f(y)) \wedge \neg\, r(f(f(z)), f(a))] \vee \forall\, y\, \forall\, z\, [r(f(f(z)), f(a)) \wedge \neg\, r(v, f(y))] \vee r(a, f(x))]) =$$

$$\bigvee_{p \in A} \bigwedge_{q \in A} [\bigvee_{t \in A} \bigvee_{u \in A} [r'^{\mathcal{A}}(p, f^{\mathcal{A}}(t)) \wedge \neg\, r'^{\mathcal{A}}(f^{\mathcal{A}}(f^{\mathcal{A}}(u)), f^{\mathcal{A}}(a^{\mathcal{A}}))] \vee$$

$$\bigwedge_{w \in A} \bigwedge_{m \in A} [r'^{\mathcal{A}}(f^{\mathcal{A}}(f^{\mathcal{A}}(m)), f^{\mathcal{A}}(a^{\mathcal{A}})) \wedge \neg\, r'^{\mathcal{A}}(p, f^{\mathcal{A}}(w))] \vee r'^{\mathcal{A}}(a^{\mathcal{A}}, f^{\mathcal{A}}(q))]),$$

unde am notat cu $r'^{\mathcal{A}} : A^2 \rightarrow \mathcal{L}_2$ operaţia binară de rezultat boolean asociată relaţiei binare $r^{\mathcal{A}} \subseteq A^2$: pentru orice $d, e \in A$, $r'^{\mathcal{A}}(d, e) = \begin{cases} 1, & \text{dacă } (d, e) \in r^{\mathcal{A}} \\ 0, & \text{dacă } (d, e) \notin r^{\mathcal{A}}. \end{cases}$ Am folosit faptul că, de exemplu, $V(r(p, f(y)) \wedge \neg\, r(f(f(z)), f(a))) = \{y, z\}$, aşadar, pentru orice $t, u \in A$, $\left(s\begin{bmatrix} z \\ u \end{bmatrix}\right)\begin{bmatrix} y \\ t \end{bmatrix}(r'^{\mathcal{A}}(p, f^{\mathcal{A}}(t)) \wedge \neg\, r'^{\mathcal{A}}(f^{\mathcal{A}}(f^{\mathcal{A}}(u))) = r'^{\mathcal{A}}(p, f^{\mathcal{A}}(t)) \wedge \neg\, r'^{\mathcal{A}}(f^{\mathcal{A}}(f^{\mathcal{A}}(u)),$ prin urmare:

$$s(\exists\, y\, \exists\, z\, [r(p, f(y)) \wedge \neg\, r(f(f(z)), f(a))]) = \bigvee_{t \in A} \bigvee_{u \in A} \left(s\begin{bmatrix} z \\ u \end{bmatrix}\right)\begin{bmatrix} y \\ t \end{bmatrix}(r'^{\mathcal{A}}(p, f^{\mathcal{A}}(t)) \wedge \neg\, r'^{\mathcal{A}}(f^{\mathcal{A}}(f^{\mathcal{A}}(u)).$$

Variabilele y şi z, care apar cuantificate şi existenţial, şi universal, servesc la parcurgerea mulţimii A (aşadar cu perechea de variabile (y,z) se parcurge mulţimea $A^2 = A \times A$). Cele două parcurgeri sunt independente, aşadar putem redenumi prima pereche de variabile (y,z) în (y',z'):

$$\varepsilon \mathrel{\models\!\!\!\dashv} \exists\, v\, \forall\, x\, [\exists\, y'\, \exists\, z'\, [r(v,f(y')) \wedge \neg\, r(f(f(z')),f(a))] \vee \forall\, y\, \forall\, z\, [r(f(f(z)),f(a)) \wedge \neg\, r(v,f(y))] \vee r(a,f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\exists\, y'\, \exists\, z'\, [[r(v,f(y')) \wedge \neg\, r(f(f(z')),f(a))] \vee \forall\, y\, \forall\, z\, [r(f(f(z)),f(a)) \wedge \neg\, r(v,f(y))]] \vee r(a,f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, [\exists\, y'\, \exists\, z'\, \forall\, y\, \forall\, z\, [[r(v,f(y')) \wedge \neg\, r(f(f(z')),f(a))] \vee [r(f(f(z)),f(a)) \wedge \neg\, r(v,f(y))]] \vee r(a,f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, \exists\, y'\, \exists\, z'\, \forall\, y\, \forall\, z\, [[r(v,f(y')) \wedge \neg\, r(f(f(z')),f(a))] \vee [r(f(f(z)),f(a)) \wedge \neg\, r(v,f(y))] \vee r(a,f(x))] \mathrel{\models\!\!\!\dashv}$$

$$\exists\, v\, \forall\, x\, \exists\, y'\, \exists\, z'\, \forall\, y\, \forall\, z\, [[r(v,f(y')) \vee r(f(f(z)),f(a)) \vee r(a,f(x))] \wedge [r(v,f(y')) \vee \neg\, r(v,f(y)) \vee r(a,f(x))] \wedge$$

$$[\neg\, r(f(f(z')),f(a)) \vee r(f(f(z)),f(a)) \vee r(a,f(x))] \wedge [\neg\, r(f(f(z')),f(a)) \vee \neg\, r(v,f(y)) \vee r(a,f(x))]].$$

Acum înlocuim variabilele cuantificate existenţial cu funcţii Skolem:

- variabila v cu o constantă Skolem c;
- variabilele y' şi z' cu funcţii Skolem unare g şi h (întrucât acestea depind de variabila x, cuantificată universal, care le precede):

$$\xi \overset{\text{notaţie}}{=} \forall\, x\, \forall\, y\, \forall\, z\, [[r(c,f(g(x))) \vee r(f(f(z)),f(a)) \vee r(a,f(x))] \wedge [r(c,f(g(x))) \vee \neg\, r(c,f(y)) \vee r(a,f(x))] \wedge$$

$$[\neg\, r(f(f(h(x))),f(a)) \vee r(f(f(z)),f(a)) \vee r(a,f(x))] \wedge [\neg\, r(f(f(h(x))),f(a)) \vee \neg\, r(c,f(y)) \vee r(a,f(x))]].$$

ξ este o formă Skolem a lui ε.

Amintesc semnificaţia funcţiilor Skolem: $\mathcal{A} \vDash \varepsilon$ (algebra $\mathcal{A}$ satisface enunţul ε), adică $||\varepsilon||_{\mathcal{A}} = 1$, ddacă, adăugând la signatură un simbol de constantă c şi două simboluri de operaţii unare g şi h, există o constantă $c^{\mathcal{A}} \in A$ şi două funcţii $g^{\mathcal{A}} : A \to A$ şi $h^{\mathcal{A}} : A \to A$ astfel încât algebra $\mathcal{A}' = (A; f^{\mathcal{A}}, g^{\mathcal{A}}, h^{\mathcal{A}}; r^{\mathcal{A}}; a^{\mathcal{A}}, c^{\mathcal{A}})$ satisface enunţul ξ. Aşadar ε e satisfiabil ddacă ξ e satisfiabil.

Forma clauzală a lui ξ, adică mulţimea de clauze corespunzătoare lui ξ, este următoarea – amintesc că trebuie să redenumim variabilele astfel încât mulţimile variabilelor care apar în clauze diferite să fie disjuncte:

$$\{r(c,f(g(x))),\ r(f(f(z)),f(a)),\ r(a,f(x))\},\ \{r(c,f(g(x'))),\ \neg\, r(c,f(y)),\ r(a,f(x'))\}$$

$$\{\neg\, r(f(f(h(x''))),f(a)),\ r(f(f(z'')),f(a)),\ r(a,f(x''))\},\ \{\neg\, r(f(f(h(x'''))),f(a)),\ \neg\, r(c,f(y'')),\ r(a,f(x'''))\}.$$

Dintre aceste clauze, $\{\neg\, r(f(f(h(x''))),f(a)),\ r(f(f(z'')),f(a)),\ r(a,f(x''))\}$ şi $\{r(c,f(g(x'))),\ \neg\, r(c,f(y)),\ r(a,f(x'))\}$ sunt clauze triviale, pentru că formulele atomice $r(f(f(h(x''))),f(a))$ şi $r(f(f(z'')),f(a))$ unifică, având $\{z''/h(x'')\}$ drept un cel mai general unificator, şi formulele atomice $r(c,f(g(x')))$ şi $r(c,f(y))$ unifică, având $\{y/g(x')\}$ drept un cel mai general unificator. Aceste formule atomice unifică privite ca termeni cu o operaţie dominantă r, de rezultat boolean, asociată relaţiei r, în modul în care, mai sus, am asociat operaţia $r'^{\mathcal{A}}$ relaţiei $r^{\mathcal{A}}$ – mai precis, adăugând în signatură un simbol de operaţie binară r căruia, în algebrele în care interpretăm aceste formule, îi corespunde o operaţie binară având rezultatul boolean (astfel că valorile booleene trebuie adăugate la mulţimile de elemente ale acelor algebre), definită, în funcţie de relaţia din acele algebre corespunzătoare simbolului de relaţie r la fel ca operaţia $r'^{\mathcal{A}}$ de mai sus în funcţie de relaţia $r^{\mathcal{A}}$; amintesc că, în acest mod, predicatele, adică relaţiile, în Prolog, devin operaţii de rezultat boolean, astfel că putem imbrica predicatele la fel cum imbricăm orice operatori, iar formulele (fără cuantificatori) devin termeni.

Algoritmul Davis–Putnam are doar un rol de exersare aici, pentru că, după cum ne amintim din curs, nu dă întotdeauna răspunsul corect, spre deosebire de cazul în care Prologul aplică rezoluţia.

Eliminăm clauzele triviale şi obţinem următoarea mulţime de clauze, care are o singură derivare prin rezoluţie:

$$\dfrac{\dfrac{\{r(c,f(g(x))),\ \cancel{r(f(f(z)),f(a))},\ r(a,f(x))\},\ \{\neg\, \cancel{r(f(f(h(x'''))),f(a))},\ \neg\, r(c,f(y'')),\ r(a,f(x'''))\} \quad \text{(cu unificatorul } \{z/h(x''')\})}{\{r(c,f(g(x))),\ r(a,f(x)),\ \neg\, r(c,f(y'')),\ r(a,f(x'''))\}}}{\emptyset}$$

Exercițiul 2. Considerăm un limbaj de ordinul I conţinând un simbol de operaţie unară f, un simbol de constantă a, un simbol de relaţie binară r şi simbol de relaţie unară q. Considerăm şi trei variabile distincte x, y şi z. Să se aplice algoritmul Davis–Putnam următoarei mulţimi de clauze:

$$\{q(f(a)), r(x, f(x))\},\ \{\neg\, q(f(y)), r(y, y)\},\ \{q(f(z))\}.$$

Rezolvare: Nu putem alege, pentru a efectua un pas al algoritmului DP, decât formula atomică $q(f(z))$. Un cel mai general unificator pentru $q(f(a))$, $q(f(y))$ şi $q(f(z))$ este: $\{y/a, z/a\}$.

$$\frac{\dfrac{\{q(f(a)), r(x, f(x))\},\ \{\neg\, q(f(y)), r(y, y)\},\ \{q(f(z))\}}{\{r(x, f(x)), r(a, a)\},\ \{r(a, a)\}}}{\emptyset}$$

Exercițiul 3. Considerăm următorul program în Prolog:
$add(X, 0, X)$.
$add(X, succ(Y), succ(Z))$:- $add(X, Y, Z)$.
şi interogările:
?- $add(succ(succ(0)), succ(succ(0)), U)$.
?- $add(U, succ(0), succ(U))$.
?- $add(U, W, succ(succ(succ(0))))$.
?- $add(U, 0, succ(0))$, $add(succ(U), U, W)$.
Să se scrie arborii de derivare prin rezoluţie pentru aceste scopuri.

Rezolvare: Avem, în signatură, o constantă 0, o operaţie unară $succ$ şi o relaţie ternară add.

Faptul $add(X, 0, X)$ corespunde enunţului $\forall X\, add(X, 0, X)$, cu forma clauzală $\{add(X, 0, X)\}$ (clauză Horn).

Regula $add(X, succ(Y), succ(Z))$:- $add(X, Y, Z)$ corespunde enunţului:

$$\forall X\, \forall Y\, \forall Z\, [add(X, Y, Z) \rightarrow add(X, succ(Y), succ(Z))] \mathrel{\models\!\!\!\mid} \forall X\, \forall Y\, \forall Z\, [\neg\, add(X, Y, Z) \vee add(X, succ(Y), succ(Z))],$$

cu forma clauzală $\{\neg\, add(X, Y, Z), add(X, succ(Y), succ(Z))\}$ (clauză Horn).

Baza de cunoştinţe corespunde conjuncţiei celor două enunţuri de mai sus, având forma clauzală:

$$\{add(V, 0, V)\},\ \{\neg\, add(X, Y, Z), add(X, succ(Y), succ(Z))\}$$

(mulţime de clauze Horn). Am redenumit variabila din prima clauză, pentru a face mulţimile variabilelor din cele două clauze disjuncte.

Scopul $add(succ(succ(0)), succ(succ(0)), U)$ corespunde enunţului $\exists U\, add(succ(succ(0)), succ(succ(0)), U)$, a cărui negaţie este echivalentă semantic cu enunţul $\forall U\, \neg\, add(succ(succ(0)), succ(succ(0)), U)$, având forma clauzală: $\{\neg\, add(succ(succ(0)), succ(succ(0)), U)\}$ (clauză scop).

Scopul $add(U, succ(0), succ(U))$ corespunde enunţului $\exists U\, add(U, succ(0), succ(U))$, a cărui negaţie este echivalentă semantic cu enunţul $\forall U\, \neg\, add(U, succ(0), succ(U))$, având forma clauzală: $\{\neg\, add(U, succ(0), succ(U))\}$ (clauză scop).

Scopul $add(U, W, succ(succ(succ(0))))$ corespunde enunţului: $\exists U\, \exists W\, add(U, W, succ(succ(succ(0))))$, a cărui negaţie este echivalentă semantic cu enunţul: $\forall U\, \forall W\, \neg\, add(U, W, succ(succ(succ(0))))$, având forma clauzală: $\{\neg\, add(U, W, succ(succ(succ(0))))\}$ (clauză scop).

Scopul compus $add(U, 0, succ(0)), add(succ(U), U, W)$ corespunde enunţului:

$$\exists U\, \exists W\, [add(U, 0, succ(0)) \wedge add(succ(U), U, W)],$$

a cărui negaţie este echivalentă semantic cu enunţul:

$$\forall U\, \forall W\, [\neg\, add(U, 0, succ(0)) \vee \neg\, add(succ(U), U, W)],$$

având forma clauzală:

$$\{\neg\, add(U, 0, succ(0)), \neg\, add(succ(U), U, W)\} \text{ (clauză scop).}$$

A se vedea arborii de derivare în fişierul *s7plarbderiv.pdf*.