Logică pentru informatică - Săptămâna 11 Forme normale ale formulelor de ordinul I - Partea a II-a

Ștefan Ciobâcă

January 14, 2018

1 Forma normală Skolem

Definiția 1.1 (FNS)**.** *O formulă* φ este în formă normală Skolem *(prescurtat, FNS) dacă*

$$\varphi = \forall x_1 . \ldots . \forall x_n . \varphi',$$

unde:

1. φ' *nu conține cuantificatori și*
2. *free*$(\varphi') \subseteq \{x_1, \ldots, x_n\}$.

Observație 1.1. *O formulă aflată în FNS este obligatoriu închisă, deoarece toate variabilele libere ale formulei* φ' *sunt cuantificate universal în* φ *(datorită condiției* 2*), și deci nu mai pot exista variabile libere în* φ*.*

Exemplul 1.1. *În continuare, vom lucra peste signatura* $\Sigma = (P, Q, R, f, i, e)$*, unde* P, Q, R *sunt simboluri predicative de aritate* $2, 1$ *și respectiv* 3*,* f, i *sunt simboluri funcționale de aritate* 2 *și respectiv* 1*, iar* e *este simbol constante.*

Exemple de formule în FNS:

$$\forall x.P(x, i(e)) \qquad \forall x.\forall y.\Big(P(f(x, e), y) \land \neg(R(x, i(f(y, y)), e) \lor Q(y))\Big)$$

Exemple de formule care nu sunt în FNS:

$$\exists x.P(x, x) \qquad \forall x.\Big(Q(e) \land \neg(Q(x) \lor Q(y))\Big) \qquad Q(e) \land \forall x.Q(x)$$

Teorema 1.1 (Teorema de aducere în FNS)**.** *Pentru orice formulă* $\varphi \in$ `LP1`*, există o formulă* $\varphi' \in$ `LP1` *astfel încât:*

1. φ' *este în formă normală Skolem;*

2. φ și φ' sunt echisatisfiabile.

Schiță de demonstrație. 1. Calculăm o formulă φ_1, aflată în FNP și echivalentă cu formula φ (folosind Teorema de aducere în FNP din cursul precedent);

2. Calculăm o formulă φ_2, închiderea existențială a lui φ_2 (φ_2 va fi echisatisfabilă cu φ_1 și deci cu φ);

3. Aplicăm în mod repetat lema de Skolemizare pe formula φ_2, lemă prezentată mai jos.

Rezultatul va fi o formulă aflată în FNS, echisatisfabilă cu formula de la care am plecat.

□

Lema 1.1 (Skolem)**.** *Fie $\varphi = \forall x_1.\forall x_2.\ldots.\forall x_k.\exists x.\varphi'$, unde $k \geq 0$, $\varphi' \in$ `LP1` (φ' poate conține alți cuantificatori). Cu alte cuvinte, există k cuantificatori universali înainte de primul cunatificator existențial.*

Fie $f \in \mathcal{F}_k$ un simbol funcțional de aritate k care nu apare în φ (un simbol funcțional proaspăt – engl. fresh).

Avem că φ este echisatisfiabilă cu

$$\forall x_1.\ldots\forall x_k.(\sigma^\flat(\varphi')),$$

unde $\sigma = \{x \mapsto f(x_1, \ldots, x_k)\}$.

Schiță de demonstrație. Implicația directă:

Presupunem că există o structură S și o atribuire α astfel încât $S, \alpha \models \varphi$.

Găsim o structură S' astfel încât $S', \alpha \models \forall x_1.\forall x_2.\ldots\forall x_k.(\sigma^\flat(\varphi'))$.

Atenție! Structura S' este peste o signatură mai bogată decât signatura structurii S (apare simbolul funcțional f de aritate k, care este nou).

Implicația inversă:

Presupunem că există o structură S' și o atribuire α astfel încât $S', \alpha \models \forall x_1.\forall x_2.\ldots\forall x_k.(\sigma^\flat(\varphi'))$.

Găsim o structură S astfel încât $S, \alpha \models \varphi$.

□

Exercițiul 1.1. *Completați demonstrația de mai sus.*

Exemplul 1.2. *Calculăm o formă normală Skolem pentru formula*

$$\varphi = \forall x.\exists y.\forall z.\exists z'.(P(x, y) \leftrightarrow P(z, z')).$$

Prin Lema 1.1, avem că φ este echisatisfiabilă cu

$$\varphi_1 = \forall x.\forall z.\exists z'.(P(x, g(x)) \leftrightarrow P(z, z')),$$

unde g este un simbol funcțional nou, de aritate 1.

Aplicând din nou Lema 1.1, avem că formula φ_1 este echisatisfiabilă cu

$$\varphi_2 = \forall x.\forall z.(P(x, g(x)) \leftrightarrow P(z, h(x, z))),$$

unde h este un simbol funcțional nou, de aritate 2.

În concluzie, φ_2 este în FNS și est echisatisfiabilă cu φ, deci este o formă normală Skolem a formulei φ.

Observație 1.2. *Signatura formei normale Skolem a unei formule este mai bogată decât signature formulei de la care am plecat, din cauza adăugării simbolurilor Skolem.*

2 Forma normală conjunctivă

Definiția 2.1 (Literal)**.** *O formulă $\varphi \in$ `LP1` se numește* literal *dacă există un simbol predicativ $P \in \mathcal{P}_n$ de aritate $n \geq 0$ și n termeni $t_1, \ldots, t_n \in \mathcal{T}$ astfel încât*

$$\varphi = P(t_1, \ldots, t_n) \text{ sau } \varphi = \neg P(t_1, \ldots, t_n).$$

Cu alte cuvinte, un literal este sau o formulă atomică, sau negația unei formule atomice.

Exemplul 2.1. *Exemple de literali:*

$$P(x, x) \qquad \neg P(x, i(y)) \qquad \neg R(a, b, c) \qquad \neg P(x, x) \qquad Q(i(x))$$

$$R(a, f(x), b)$$

Exemple de formule care nu sunt literali:

$$P(x, y) \wedge P(x, y) \qquad \neg\neg P(x, x) \qquad \forall x.P(x, x)$$

Definiția 2.2 (Clauză)**.** *O formulă $\varphi \in$ `LP1` se numește* clauză *dacă există n literali $\varphi_1, \ldots, \varphi_n \in$ `LP1` astfel încât*

$$\varphi = \varphi_1 \vee \varphi_2 \vee \ldots \vee \varphi_n.$$

Exemplul 2.2.

$$P(x, x) \vee R(x, e, y) \vee \neg P(x, f(e, y)) \qquad P(x, x) \qquad \square \qquad P(x, x)$$

$$P(x, x) \vee R(x, f(e, e), y) \vee \neg P(x, f(e, y)) \vee \neg P(e, i(x))$$

Observație 2.1. *Un caz particular este reprezentat de clauza vidă, notată $\square$, care este disjuncția a 0 literali. Clauza vidă este o formulă nesatisfiabilă.*

Un alt caz particular este reprezentat de literali. Orice literal este și clauză, fiind considerat disjuncția unui singur literal.

Definiția 2.3 (FNC)**.** *O formulă φ este* în formă normală clauzală *(sau, echivalent,* în formă normală conjunctivă*) dacă există n clauze $\varphi_1, \ldots, \varphi_n$ astfel încât*

$$\varphi = \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n.$$

Exemplul 2.3. *Următoarele formule sunt în FNC:*

$$(P(x, x) \vee Q(x)) \wedge (\neg P(x, y) \vee R(x, y, e))$$

$$(P(x, y) \vee Q(i(x)) \vee \neg Q(e)) \wedge (\neg P(x, x)) \wedge (\neg Q(f(z, z)) \vee R(x, z))$$

3 Forma normală Skolem clauzală

Definiția 3.1. *O formulă φ* este în formă normală Skolem clauzală *(prescurtat FNSC) dacă*

1. *φ este în formă normală Skolem și*
2. *φ' este în formă normală clauzală, unde: $\varphi = \forall x_1. \ldots. \forall x_n.\varphi'$, unde φ' nu are cuantificatori (cu alte cuvinte, φ' este subformula obținută din φ prin ștergerea cuantificatorilor).*

Exemplul 3.1. *Exemple de formule în FNSC:*

$$\forall x.\forall y.\Big((P(x, x) \vee \neg Q(i(x)) \wedge (P(e, y) \vee \neg Q(e))\Big)$$

$$\forall x.\forall y.\Big(Q(e) \wedge (\neg R(x, e, y) \vee Q(i(y)))\Big)$$

$$\forall x.\forall y.\forall z.(P(x, y) \wedge (Q(x) \vee R(x, y, z)) \wedge \neg Q(x)).$$

Exemple de formule care nu sunt în FNSC:

$$\exists x.Q(x) \qquad \forall x.\Big(Q(e) \wedge (\neg R(x, y, z) \vee Q(y))\Big)$$

$$\forall x.\forall y.\Big(Q(e) \wedge \neg(Q(x) \vee Q(y))\Big)$$

Teorema 3.1. *Pentru orice formulă $\varphi \in$ `LP1` aflată în FNS există o formulă $\varphi' \in$ `LP1` astfel încât:*

1. *φ' este în FNSC și*
2. *$\varphi \equiv \varphi'$.*

Schiță de demonstrație. Se aplică de la stânga la dreapta următoarele echivalențe:

1. $\varphi_1 \leftrightarrow \varphi_2 \equiv (\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$;

2. $\varphi_1 \rightarrow \varphi_2 \equiv \neg\varphi_1 \vee \varphi_2$;

3. $\neg\neg\varphi \equiv \varphi$;

4. $\neg(\varphi_1 \vee \varphi_2) \equiv \neg\varphi_1 \wedge \neg\varphi_2$;

5. $\neg(\varphi_1 \wedge \varphi_2) \equiv \neg\varphi_1 \vee \neg\varphi_2$;

6. $\varphi_1 \vee (\varphi_2 \wedge \varphi_3) \equiv (\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)$.

De asemenea, se permite folosirea liberă a asociativității și comutativității conectorilor $\vee$ și $\wedge$.

Primele două echivalențe elimină orice folosire a cuantificatorilor $\leftrightarrow$ și $\rightarrow$.

Următoarele trei echivalențe se asigură că negațiile pot fi aplicate doar formulelor atomice.

Ultima echivalență se asigură că $\vee$-ul nu apare peste $\wedge$.

În final, vom avea o formulă în care spre rădăcină (după cunatificatorii universali) avem $\wedge$, urmat de un strat de $\vee$, urmat de $\neg$, urmat de formule atomice, ceea ce înseamnă că formula rezultată este în FNSC.

□

4 Un exemplu de aducere în FNSC

Ne interesează să stabilim validitatea formulei

$$\varphi = \Big(\forall x.\big(Q(x) \leftrightarrow \neg Q(i(x))\big)\Big) \rightarrow \Big(\forall x.\big(Q(x) \rightarrow Q(i(i(x)))\big)\Big).$$

Este ușor de văzut că o formulă este validă dacă și numai dacă negația ei este nesatisfiabilă. Astfel încât, pentru a stabili că φ este validă, este suficient să arătăm că

$$\neg\varphi = \neg\Bigg(\Big(\forall x.\big(Q(x) \leftrightarrow \neg Q(i(x))\big)\Big) \rightarrow \Big(\forall x.\big(Q(x) \rightarrow Q(i(i(x)))\big)\Big)\Bigg)$$

este nesatisfiabilă.

Vom calcula o FNSC clauzală a formulei $\neg\varphi$, FNSC despre care știm că este echisatisfiabilă cu formula $\neg\varphi$. Primul pas este să găsim o FNP:

$$
\begin{aligned}
\neg\varphi = \quad & \neg\Big(\big(\forall x.(Q(x) \leftrightarrow \neg Q(i(x)))\big) \rightarrow \big(\forall x.(Q(x) \rightarrow Q(i(i(x))))\big)\Big)\\
\equiv \quad & \neg\Big(\neg\big(\forall x.(Q(x) \leftrightarrow \neg Q(i(x)))\big) \vee \big(\forall x.(Q(x) \rightarrow Q(i(i(x))))\big)\Big)\\
\equiv \quad & \big(\neg\neg\forall x.(Q(x) \leftrightarrow \neg Q(i(x)))\big) \wedge \big(\neg\forall x.(Q(x) \rightarrow Q(i(i(x))))\big)\\
\equiv \quad & \big(\forall x.(Q(x) \leftrightarrow \neg Q(i(x)))\big) \wedge \big(\exists x.\neg(Q(x) \rightarrow Q(i(i(x))))\big)\\
\equiv \quad & \forall x.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \exists x.\neg(Q(x) \rightarrow Q(i(i(x))))\big)\\
\equiv \quad & \forall x.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \exists x'.\neg(Q(x') \rightarrow Q(i(i(x'))))\big)\\
\equiv \quad & \forall x.\exists x'.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(x') \rightarrow Q(i(i(x'))))\big).
\end{aligned}
$$

Așadar, o formulă normală prenex a formulei $\neg\varphi$ este

$$\varphi_1 = \forall x.\exists x'.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(x') \rightarrow Q(i(i(x'))))\big).$$

Continuăm prin găsirea unei FNS a formulei φ_1, prin aplicarea lemei de skolemizare, și folosind un simbol Skolem proaspăt g, de aritate 1:

$$
\begin{aligned}
\varphi_1 = \quad & \forall x.\exists x'.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(x') \rightarrow Q(i(i(x'))))\big)\\
\text{echisatisfiabilă cu} \quad & \forall x.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(g(x)) \rightarrow Q(i(i(g(x)))))\big).
\end{aligned}
$$

Așadar, o FNS a formulei $\neg\varphi$ este formula $\varphi_2 = \forall x.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(g(x)) \rightarrow Q(i(i(g(x)))))\big)$. Continuăm, pentru a găsi o FNSC a formulei φ_2:

$$
\begin{aligned}
\varphi_2 = \quad & \forall x.\big((Q(x) \leftrightarrow \neg Q(i(x))) \wedge \neg(Q(g(x)) \rightarrow Q(i(i(g(x)))))\big).\\
\equiv \quad & \forall x.\big((Q(x) \rightarrow \neg Q(i(x))) \wedge (\neg Q(i(x)) \rightarrow Q(x)) \wedge \neg(\neg Q(g(x)) \vee Q(i(i(g(x)))))\big).\\
\equiv \quad & \forall x.\big((\neg Q(x) \vee \neg Q(i(x))) \wedge (\neg\neg Q(i(x)) \vee Q(x)) \wedge (\neg\neg Q(g(x)) \wedge \neg Q(i(i(g(x)))))\big).\\
\equiv \quad & \forall x.\big((\neg Q(x) \vee \neg Q(i(x))) \wedge (Q(i(x)) \vee Q(x)) \wedge (Q(g(x)) \wedge \neg Q(i(i(g(x)))))\big).\\
\equiv \quad & \forall x.\big((\neg Q(x) \vee \neg Q(i(x))) \wedge (Q(i(x)) \vee Q(x)) \wedge (Q(g(x))) \wedge (\neg Q(i(i(g(x)))))\big).
\end{aligned}
$$

5 Rezoluția de bază

Pentru a testa satisfiabilitatea unei formule aflate în FNSC, putem folosi sistemul deductiv descris în secțiunea aceasta.

Definiția 5.1. *Un termen $t \in \mathcal{T}$ se numește* termen de bază *dacă $vars(t) = \emptyset$.*

În engleză, termen de bază = ground term.

Definiția 5.2. *O substituție* $\sigma = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ *se numește* substitutie de bază *dacă* $t_1, \ldots, t_n$ *sunt termeni de bază.*

În engleză, substituție de bază = ground substitution.

Definiția 5.3. *Rezoluția de bază este un sistem deductiv*[1] *pentru clauze, cu următoarea regulă de inferență, numită rezoluție de bază:*

$$\frac{C_1 \vee P(t_1, \ldots, t_n) \qquad C_2 \vee \neg P(t'_1, \ldots, t'_n)}{\sigma_1^\flat(C_1) \vee \sigma_2^\flat(C_2)} \quad \begin{array}{l} \sigma_1 \text{ E SUBST. DE BAZĂ} \\ \sigma_2 \text{ E SUBST. DE BAZĂ} \\ \sigma_1^\sharp(t_i) = \sigma_2^\sharp(t'_i) \text{ PENTRU ORICE } 1 \leq i \leq n \end{array}$$

Teorema 5.1 (Teorema rezoluției de bază)**.** *O formulă* φ *aflată în FNSC este nesatisfiabilă ddacă se poate obține* $\square$ *prin rezoluție de bază pornind de la clauzele din* φ*.*

Exemplul 5.1. *Vom arăta, folosind rezoluția de bază, că formula* $\forall x.\Big(\big(\neg Q(x) \vee \neg Q(i(x))\big) \wedge \big(Q(i(x)) \vee Q(x)\big) \wedge \big(Q(g(x))\big) \wedge \big(\neg Q(i(i(g(x))))\big)\Big)$ *este nesatisfiabilă folosind rezoluția de bază:*

1. $\neg Q(x) \vee \neg Q(i(x))$*;*
2. $Q(i(x)) \vee Q(x)$*;*
3. $Q(g(x))$*;*
4. $\neg Q(i(i(g(x))))$*;*
5. $\neg Q(i(g(e)))$ *(3,1,*$\sigma_1 = \{x \mapsto e\}$*,*$\sigma_2 = \{x \mapsto g(e)\}$*,* $\sigma_1^\flat(Q(g(x))) = \sigma_2^\flat(Q(x))$*);*
6. $Q(i(i(g(e))))$ *(2,5,*$\sigma_1 = \{x \mapsto i(g(e))\}$*,* $\sigma_2 = \{\}$*,* $\sigma_1^\flat(Q(x)) = \sigma_2^\flat(Q(i(g(e))))$*);*
7. $\square$ *(6, 4,* $\sigma_1 = \{\}$*,* $\sigma_2 = \{x \mapsto e\}$*,* $\sigma_1^\flat(Q(i(i(g(e))))) = \sigma_2^\flat(Q(i(i(g(x)))))$*).*

Deoarece am ajuns la clauza vidă, concluzionăm prin Teorema 5.1 că formula $\forall x.\Big(\big(\neg Q(x) \vee \neg Q(i(x))\big) \wedge \big(Q(i(x)) \vee Q(x)\big) \wedge \big(Q(g(x))\big) \wedge \big(\neg Q(i(i(g(x))))\big)\Big)$ *este nesatisfiabilă.*

Dar formula de mai sus este o FNSC a formulei $\neg\varphi$*, deci cele două sunt echisatisfiabile. Deci* $\neg\varphi$ *este de asemenea nesatisfiabilă. Deci formula cu care am pornit,*

$$\varphi = \Big(\forall x.\big(Q(x) \leftrightarrow \neg Q(i(x))\big)\Big) \rightarrow \Big(\forall x.\big(Q(x) \rightarrow Q(i(i(x)))\big)\Big),$$

este validă.

[1] De fapt, sistemul deductiv mai conține o regulă, numită factorizare, pe care o vom vedea mai târziu