Capitolul 3

Logica (calculul) cu predicate de ordinul I

După cum am punctat în finalul **Capitolului 2**, suntem nevoiţi să extindem limbajul (logica) folosit(ă) până în prezent, deşi rezultatele privind complexitatea (timp) a algoritmilor de testare a satisfiabilităţii formulelor **LP** sunt deja descurajatoare. Principalul argument este acela că **lumea reală nu poate fi modelată satisfăcător şi simplu prin formule logice care utilizează doar variabile propoziţionale şi conectori**. De aceea principala modificare va fi de *natură sintactică*, manifestată prin adăugarea *cuantificatorilor logici* (există şi pentru orice), împreună cu introducerea *variabilelor*, *constantelor*, *simbolurilor funcţionale (de aritate mai mare decât unu)* şi a *simbolurilor predicative*. Din acest punct de vedere, ***limbajul logicii cu predicate de ordinul I*** va conduce ***sintactic*** la o mulţime de formule, notată **LP1**, care va include strict clasa **LP**. ***Semantica*** pentru **LP1** va fi mai complexă, având însă la bază acelaşi concept de ***structură (asignare, interpretare)*** şi fiind „consistentă" cu semantica **LP**. De altfel, **pentru LP1, dacă nu vom folosi concepte prezentate în mod explicit ca fiind noi, toate notaţiile, noţiunile, rezultatele, etc., vor fi identice cu cele introduse pentru LP**.

§1. Sintaxa logicii cu predicate de ordinul I

Prin urmare, pentru a mări *puterea de exprimare*, în sensul informal precizat deja, a logicii folosite până în prezent (**LP**), vom începe cu anumite transformări de **natură sintactică**. Mai precis, pentru a construi mulţimea de formule a logicii cu predicate de ordinul I, **LP1**, vom porni cu următoarele mulţimi de simboluri:

- $\mathcal{X} = \{x_1, x_2, \ldots\}$: o mulţime cel mult numărabilă de **variabile funcţionale,** sau, pe scurt, **variabile**.
- $\mathcal{P} = \{\mathcal{P}_0, \mathcal{P}_1, \ldots\}$: o mulţime cel mult numărabilă de **simboluri predicative** (sau **predicate**, sau **relaţii**), cu arităţi. La rândul său, fiecare $\mathcal{P}_i$ este o mulţime cel mult numărabilă de **predicate de aritate i** ($i \in \mathbf{N}$). Elementele lui $\mathcal{P}_0$ se mai numesc şi **variabile predicative**.
- $\mathcal{F} = \{\mathcal{F}_0, \mathcal{F}_1, \ldots\}$: o mulţime cel mult numărabilă de **simboluri funcţionale** (sau **funcţii**) cu arităţi, fiecare $\mathcal{F}_i$ fiind o mulţime cel mult numărabilă de **funcţii de aritate i** ($i \in \mathbf{N}$). Elementele lui $\mathcal{F}_0$ se numesc şi **constante (funcţionale)**.
- $C_1 = \{\rceil, \vee, \wedge\}$: o mulţime de **conectori logici (conective logice)**, la care se pot adăuga, opţional, şi alte simboluri cum ar fi $\rightarrow$, $\leftrightarrow$, etc.
- $C_2 = \{(\forall x) \mid x \in \mathcal{X}\} \cup \{(\exists\, x) \mid x \in X\}$: o mulţime de **cuantificatori (cuantori)**, **universali**, respectiv **existenţiali** ($(\forall x)$ se citeşte *„pentru fiecare x”*, sau *„pentru oricare (orice) x”*, iar $(\exists\, x)$ – *„există x”*, *„există măcar un x”*, etc.). Pentru început este suficient să considerăm doar cuantorii universali.

Ca şi în cazul **LP**, vom porni cu alfabetul total, adică reuniunea mulţimilor precedente, împreună cu parantezele rotunde şi virgula (pe scurt, P):

$$\mathcal{Alf} = \mathcal{X} \cup (\bigcup_{i=0}^{\infty} \mathcal{P}_i) \cup (\bigcup_{i=0}^{\infty} \mathcal{F}_i) \cup C_1 \cup C_2 \cup P$$

Mulţimile $\bigcup_{i=0}^{\infty} \mathcal{P}_i$ şi $\bigcup_{i=0}^{\infty} \mathcal{F}_i$ vor fi notate tot cu $\mathcal{P}$ respectiv $\mathcal{F}$, atunci când nu există confuzii.

Observaţie. Prin *„o mulţime cel mult numărabilă"* înţelegem o mulţime numărabilă, finită sau vidă. Intuitiv, variabilele funcţionale notate x vor fi *nume generice* pentru *elementele* dintr-un anumit domeniu, care va fi fixat ulterior (prin intermediul *funcţiei semantice*). Un simbol predicativ P de aritate i reprezintă o *relaţie i-ară neprecizată*, adică este un *nume generic* pentru orice funcţie cu i argumente peste acelaşi domeniu, codomeniul fiind {**a, f**}, sau **B** (într-o *interpretare* vom avea $P(<a_1, a_2, \ldots a_i>) = 1$ dacă şi numai dacă elementele a_1, a_2, ..., a_i sunt în relaţia numită P). Similar, un simbol funcţional $f \in \mathcal{F}_i$, este numele generic al oricărei funcţii de i argumente, peste acelaşi domeniu şi codomeniu. Pentru parantezele utilizate în scrierea cuantificatorilor ar trebui de fapt folosit un alt font (ca de altfel şi pentru virgulă). ■

Mulţimea formulelor, $\mathbf{LP1}_{\mathcal{Alf}}$ (indicele va fi pus în evidenţă doar atunci când lipsa sa ar putea genera confuzii) va fi definită structural, analog cu cazul **LP** (elemente mulţimilor C_1, C_2, P, exceptând virgula, pot fi considerate ca fiind operatori pe şiruri de caractere, de aritate 1 sau 2).

Definiţia 3.1 (sintaxa LP1). Fie $\mathcal{Alf}$ alfabetul fixat anterior. Atunci mulţimea **formulelor calculului cu predicate de ordinul I**, **LP1**$_{\mathcal{Alf}}$, este dată constructiv prin:

Baza. Se defineşte mulţimea **formulelor atomice**, notată cu $\mathcal{At}$, prin:

(i) $P_o \subseteq \mathcal{At}$ (variabilele predicative sunt formule atomice).

(ii) Pentru fiecare n $\in$ **N***, pentru fiecare P $\in \mathcal{P}_n$, pentru fiecare $t_1, t_2, \ldots, t_n \in \mathcal{T}$, avem $P(t_1, t_2, \ldots\ldots, t_n) \in \mathcal{At}$.

(iii) Nimic altceva nu mai este formulă atomică.

În cele de mai sus, $\mathcal{T}$ denotă mulţimea **termilor (funcţionali)**, care este la rândul ei definită constructiv astfel:

> **Baza**. $\mathcal{X} \subseteq \mathcal{T}$ şi $\mathcal{F}_o \subseteq \mathcal{T}$ (*variabilele şi constantele sunt termi*).
>
> **Pas constructiv**. Pentru fiecare n $\in$ **N***, pentru fiecare f $\in \mathcal{F}_n$, pentru fiecare $t_1, t_2, \ldots, t_n \in \mathcal{T}$, avem $f(t_1, t_2, \ldots\ldots, t_n) \in \mathcal{T}$.

Concluzionăm această etapă a definiţiei prin a „spune” că $\mathcal{At} \subseteq$ **LP1** (*formulele atomice sunt formule*).

Pas constructiv. Continuăm definirea lui **LP1**$_{\mathcal{Alf}}$ cu partea *„formule vechi din formule noi”*.

(i) Dacă F $\in$ **LP1** atunci ($\rceil$F) $\in$ **LP1**.

(ii) Dacă $F_1, F_2 \in$ **LP1** atunci $(F_1 \wedge F_2)$, $(F_1 \vee F_2) \in$ **LP1** (dacă dorim, putem introduce şi $(F_1 \to F_2)$, $(F_1 \leftrightarrow F_2) \in$ **LP1**, etc.).

(iii) Dacă F $\in$ **LP1** atunci $(\forall x)(F) \in$ **LP1** (dacă dorim, punem şi $(\exists\ x)(F) \in$ **LP1**), pentru fiecare $x \in \mathcal{X}$. ■

Ca şi în cazul **LP**, se definesc constructiv *subf(F), mulţimea subformulelor* formulei F, şi *Arb(F), arborele ataşat* lui F (cuantorii sunt operatori de aritate 1). Singura subformulă a unei formule atomice este ea însăşi şi Arb(F) va fi constituit în acest caz dintr-un simplu nod. Un term poate fi, la rândul său, privit ca un arbore (ca de altfel şi orice formulă atomică), astfel încât arborele unei formule poate fi „detaliat”, dacă înlocuim fiecare nod corespunzător unui term cu arborele ataşat acestuia (similar pentru o subformulă atomică). În definiţia precedentă considerăm ca am pus implicit şi (desigur că urmează *Nimic altceva nu mai este formulă*);

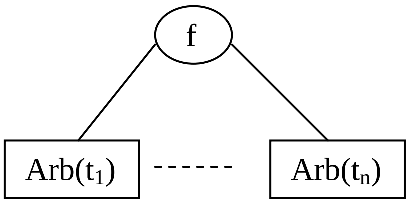
(iv) Dacă F $\in$ **LP1** atunci (F) $\in$ **LP1.**

Definiţia 3.2 (arborele ataşat unui term şi unei formule atomice). Conform definiţiei anterioare, arborele ataşat unui term $t \in \mathcal{T}$, notat Arb(t), poate fi dat constructiv prin:

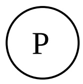
Baza. Dacă $t = c \in \mathcal{F}_0$, atunci Arb(t) este: c

Dacă $t = x \in \mathcal{X}$, atunci Arb(t) este: x

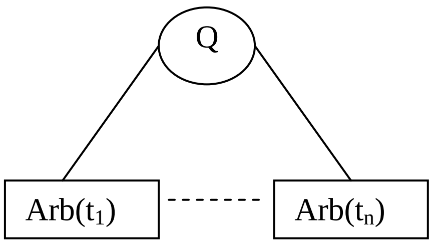
Pas constructiv. Fie $n \in$ **N***, $f \in \mathcal{F}_n$, $t_1, t_2, \ldots, t_n \in \mathcal{T}$, astfel încât $t = f(t_1, t_2, \ldots\ldots, t_n)$. Să presupunem că ştim arborii ataşaţi termilor $t_1, t_2, \ldots, t_n$, adică $Arb(t_1), Arb(t_2), \ldots, Arb(t_n)$. Atunci Arb(t) va fi:



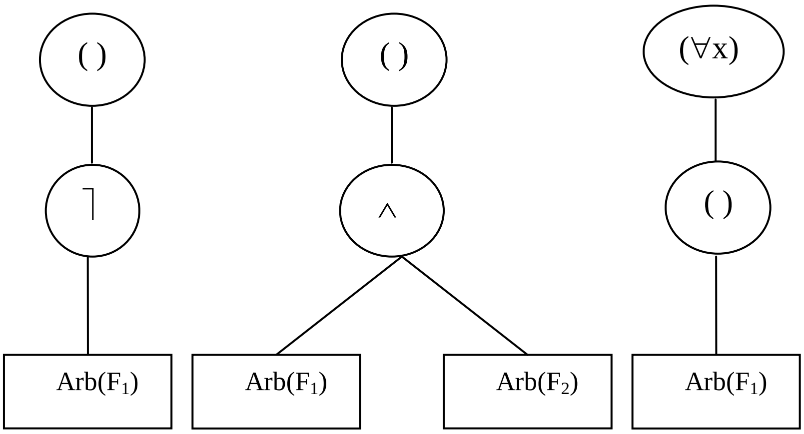
Ştiind Arb(t) pentru fiecare t $\in \mathcal{T}$, putem acum construi Arb(F) pentru fiecare F $\in \mathcal{At}$, după cum urmează.

(i) Dacă P $\in \mathcal{P}_0$, atunci Arb(P) este: 

(ii) Fie n $\in$ **N***, Q $\in$ P_n, t_1, t_2, ……, t_n $\in \mathcal{T}$ astfel încât P = Q(t_1, t_2, ……, t_n) şi să presupunem că sunt cunoscuţi Arb(t_1), Arb(t_2), … , Arb(t_n). Atunci Arb(P) va fi:



În sfârşit, fie F o formulă oarecare. Ea poate avea una din formele: ($\rceil F_1$), ($F_1 \wedge F_2$), (Arb(($F_1 \vee F_2$)) este similar cu cel pentru $\wedge$) ($\forall$x)(F_1) (similar, ($\exists$x)(F_1)), sau (F_1) (Arb(F_1) coincide cu Arb(($\rceil F_1$)), lipsind doar nodul etichetat cu $\rceil$), unde F_1, F_2 sunt tot formule (oarecare). Arborele ataşat lui F va avea în consecinţă una dintre formele:



■

Observaţie. Şi arborele pentru implică, echivalent, etc. este similar cu cel pentru şi. După cum se poate vedea, parantezele folosite la construcţia termilor şi a formulelor atomice nu apar explicit, la fel cum parantezele pentru cuantificatori sunt considerate ca făcând parte din numele acestora. Ideea este că dacă se cunosc arităţile simbolurilor funcţionale şi predicative şi dacă impunem în plus ca toate mulţimile de simboluri distincte care apar în $\mathcal{Alf}$ să fie disjuncte, atunci putem scrie $Pt_1t_2...t_n$ în loc de $P(t_1, t_2, ..., t_n)$, adică putem elimina de tot aceste paranteze (care oricum trebuiau să aibă alt font decât cele din P). Şi în cazul cuantificatorilor, în loc de $(\forall x)(F)$ putem scrie $\forall_x(F)$. Astfel, putem conveni că singurele paranteze care „merită" a fi luate în considerare atunci când apar într-o formulă sunt cele din P. ■

În cazul lui subf(F), poate este bine să subliniem faptul că subf(($\forall$x)(F_1)) = {($\forall$x)(F_1)} ∪ {(F)} ∪ subf(F_1), în rest nefiind diferenţe esenţiale faţă de cazul **LP**.

Definiţia 3.3 (apariţii libere şi legate ale variabilelor). Fie F $\in$ **LP1** şi x $\in$ $\mathcal{X}$, astfel încât x apare în F, la o poziţie oarecare j (în sens textual, stânga/dreapta, ca *literă într-un cuvânt*, apariţia menţionată *nefiind parte a numelui unui cuantificator* ($\forall$x) sau ($\exists$x)). Apariţia fixată a lui x se numeşte **legată** dacă este într-o parte (subformulă) G a unei (alte) subformule a lui F de forma G_1 = ($\forall$x)(G) (sau ($\exists$x)(G)). În restul cazurilor, apariţia considerată se numeşte **liberă**. ■

Să mai punctăm o dată faptul că folosind reprezentarea formulelor ca arbori (precum şi o definiţie corespunzătoare a noţiunii de arbore), orice apariţie a unei variabile într-o formulă poate fi definită formal într-un mod simplu (conform exemplelor şi exerciţiilor). Vom nota, pentru fiecare F $\in$ **LP1**, cu *free(F)* - mulţimea variabilelor care au apariţii libere în F, şi cu *leg(F)* – mulţimea variabilelor care au apariţii legate în F. Desigur că pentru fiecare x $\in$ X, este posibil ca x să nu apară în F, să aibă doar apariţii libere, doar apariţii legate, sau şi apariţii libere şi apariţii legate. Putem nota cu *var(F)* = free(F) ∪ leg(F). O situaţie nenaturală din punct de vedere semantic, dar posibilă sintactic, este aceea în care o variabilă x nu apare de loc în F (în sensul considerat), dar este prezent ca nume al unui cuantificator. Vom conveni să notăm mulţimea acestor variabile cu *restvar(F)* şi să includem şi această mulţime în *var(F).*

***Exerciţiul 3.1**. Definiţi constructiv leg(F) şi free(F).*

Definiţia 3.4 (închideri). O formulă F $\in$ **LP1** se numeşte **închisă** dacă nu conţine apariţii libere de variabile (altfel spus, free(F) = Ø). Pentru formula F, se numeşte **închiderea sa universală** formula $(\forall x_1)((\forall x_2)(\ \ldots((\forall x_k)(F))\ \ldots\ )$ (notată, pentru simplitate şi cu $(\forall_*)(F)$ sau chiar $(\forall F)$), unde $\{x_1, x_2, \ldots, x_k\}$ = free(F). Analog (înlocuind $\forall$ cu $\exists$) se defineşte (notează) **închiderea existenţială** a lui F. Se va numi **matricea** lui F (notată F^*) acea formulă obţinută din F prin ştergerea (sintactică, textuală) a tuturor cuantificatorilor $(\forall x)$ şi $(\exists x)$. O formulă care nu este închisă, se numeşte **deschisă** (o formulă în care var(F) = Ø se consideră a fi închisă). ■

Definiţia 3.5 (substituţii). Prin **substituţie** vom înţelege o secvenţă finită de elemente de tipul [x/t] (numite şi **substituţii elementare**), unde $x \in \mathcal{X}$, $t \in \mathcal{T}$. ■

O substituţie va avea astfel forma $s = [x_1/t_1]\bullet[x_2/t_2]\bullet \ldots \bullet[x_n/t_n]$ fiind practic un cuvânt peste alfabetul **S** = $\{[x/t] \mid x \in \mathcal{X}, t \in \mathcal{T}\}$, adică un element al lui **S*** (monoidul liber generat de **S**). **O substituţie s (ca mai sus) se aplică unei formule F**, rezultând o formulă G, notată $(F)s$, care se obţine din F prin *înlocuirea fiecărei apariţii libere* a variabilei x_1 cu termul t_1, *apoi* a fiecărei apariţii libere a variabilei x_2 cu t_2, etc.

De obicei, se utilizează doar **substituţii permise** pentru o formulă F $\in$ **LP1** dată. *Substituţia elementară [x/t] este permisă pentru F (sau, F* ***acceptă*** *[x/t])* dacă *t nu conţine variabile libere care au apariţii legate în F* (s de mai sus va fi permisă pentru F dacă va fi permisă pentru fiecare *componentă* a sa, $[x_i/t_i]$, i $\in$ [n]). Ordinea (fixată deja prin modul de scriere) aplicării substituţiilor elementare dintr-o substituţie s este esenţială în majoritatea cazurilor. O substituţie s este **normalizată (pentru F)** dacă *ordinea de aplicare a substituţiilor elementare componente nu contează*. Mai precis, s este normalizată dacă avem (F)s = (F)s', pentru fiecare s' care este obţinută din s printr-o *permutare* a componentelor acesteia. **Substituţia vidă** (ca element neutru al lui **S***), notată [], nu face desigur nici o transformare în formula F căreia îi este aplicată, adică avem (F)[] = F.

Exerciţiul 3.2*. Arătaţi că pentru fiecare F $\in$* **LP1** *şi fiecare substituţie s, permisă pentru F, care îndeplineşte condiţia că ea nu coţine un element de forma [x/t] astfel încât t nu conţine x, există măcar o substituţie s',* ***ehivalentă*** *cu s (s' este echivalentă cu s, pentru F, dacă (F)s = (F)s') şi care este normalizată.*

După cum am mai precizat, dacă nu este nevoie de o redefinire explicită, în continuare vom folosi şi alte concepte/notaţii sintactice introduse în **Capitolul 2** pentru **LP**, cum ar fi: ***literal*** (o formulă atomică sau negaţia acesteia), ***clauză***, ***clauză Horn***, ***formă normală***, etc. Ca o convenţie, cu litere latine mici de la sfîrşitul alfabetului (x, y, z, ...) vom nota variabilele. Literele latine mici de la începutul alfabetului (a, b, c, ...) vor nota constantele. Literele latine mari de la mijlocul alfabetului (P, Q, R, ...) vor nota simbolurile predicative. Literele latine mici de la mijlocul alfabetului (f, g, h, ...) vor fi rezervate pentru desemnarea simbolurilor funcţionale, iar cu F, G, H, … vom nota formulele. Dacă nu am adopta astfel de convenţii (dorim, totusi, să avem o anumită libertate de exprimare şi să folosim şi alte nume înafara celor admise de alfabetul iniţial, aşa cum am folosit şi în cazul **LP**), am avea, posibil, cazuri în care, la o primă vedere, *nu am putea distinge între o variabilă şi o constantă*. Sintactic, diferenţa dintre un nume de variabilă (element din $\mathcal{X}$) şi un nume de constantă (element al lui $\mathcal{F}_0$) este clară doar în momentul când numele respectiv apare într-un cuantificator.

Definiţia 3.6. Un term care nu conţine variabile se numeşte **term de bază**. Analog, vom avea **formule de bază, substituţii de bază**, etc. ■

Ca şi pentru **LP**, putem *renunţa la* anumite paranteze (nu uităm, sunt implicate doar elementele lui P) într-o formulă, bazându-ne pe aceeaşi idee (priorităţi acordate operatorilor, proprietăţi generale ale acestora cum ar fi asociativitatea, convenţii, etc). În cazul unui cuantor, avem nevoie, înafara priorităţii care este maximă (adică 0; standard, vom admite că $\rceil$ are prioritatea 1, $\wedge$ - 2, $\vee$ - 3, $\rightarrow$ - 4, $\leftrightarrow$ - 5, etc.; în cadrul unei secvenţe continue de operatori de aceeaşi prioritate, parantetizarea se face *la dreapta*), şi de definirea a ceea ce se numeşte

domeniul sintactic al cuantificatorului respectiv. *Domeniul sintactic pentru o apariţie a unui cuantor (să spunem că aceasta este ($\forall$x)) într-o formulă F*, reprezintă *porţiunea continuă de text* (care *este o subformulă G* alui F), care *începe* cu primul simbol de după o apariţie a cuantorului în cauză, identificat mai jos prin *subliniere* (parcurgem formula de la stânga la dreapta) şi *se termină* cu un anumit simbol ulterior (să-l identificăm tot prin *subliniere*), conform următoarelor situaţii:

- **F = ... ($\forall$x)(G') ...** . Nu sunt probleme, deoarece parantezele cerute de sintaxa dată de **Definiţia 3.1**. nu au fost eliminate. Ca observaţie, parantezele subliniate de mai sus sunt (se mai numesc şi) *corespondente* (definiţia sintactică a unei formule nu permite existenţa unei paranteze deschise fără ca ea să fie închisă ulterior, neexistând ambiguităţi nici în privinţa parantezei închise care corespunde unei anumite paranteze deschise).
- **F = ... ($\forall$x)G**. Situaţia evidenţiază faptul că *domeniul se extinde „până la sfârşitul formulei”*, dar condiţia suplimentară este aceea că *în G nu există nici o paranteză închisă care să nu aibă corespondent o paranteză deschisă situată tot în domeniu* (nu uităm că domeniul începe oricum cu primul simbol de după ($\forall$x)). Nu am mai subliniat simbolurile de început şi sfârşit a domeniului, deoarece ei sunt primul şi respectiv ultimul simbol din G.

- **F = ... (... (∀x) ...) ...** . Este pus în evidenţă cazul în care au fost eliminate parantezele iniţiale de după cuantificator, dar în subformula G care urmează după acesta există o paranteză închisă care are corespondent *înainte* de apariţia cuantorului al carui domeniu îl căutăm. Ceea ce am subliniat este **prima** paranteză cu această proprietate (practic, simbolul imediat de dinaintea ei fiind ultimul care aparţine domeniului).

Prioritatea cuantificatorilor fiind maximă, se găsesc mai întâi domeniile sintactice ale fiecărui cuantificator dintr-o formulă şi apoi se aplică regulile de prioritate legate de conectori (acestea fiind similare de fapt cu cele din **LP**). Intuitiv, dacă privim **LP1** ca un *limbaj de programare imperativ (formulele fiind programe)*, construcţiile de tipul $(\circ_1 x_1)(\circ_2 x_2) \ldots (\circ_n x_n)$, cu $\circ_1, \circ_2, \ldots, \circ_n \in \{\forall, \exists\}$, reprezintă *linia de definiţie a unei proceduri* (în care se specifică drept parametri *variabilele locale*). Procedurile n-au însă şi un sfârşit marcat explicit, astfel încât *domeniul sintactic este corpul procedurii, delimitarea lui clară fiind desigur absolut necesară pentru a cunoaşte porţiunea de text în care numele unei variabile păstrează o aceeaşi semnificaţie*. Continuând paralela, *variabilele libere din domeniul sintactic al unui cuantificator reprezintă variabilele globale,* iar *aplicarea unei substituţii ar fi un apel de procedură* (deosebirea faţă de un program imperativ fiind aceea că valorile parametrilor, adică numele cuantificatorilor din linia de definiţie, pot fi elemente *oarecare* sau *anumite* din domeniile considerate, conform *sematicii*). Să mai precizăm şi faptul ca domeniile sintactice ale diverşilor cuantificatori pot fi, datorită formei sintactice a formulelor, doar **disjuncte**, adică de tipul ... $\underline{1}$ ... $\underline{1}$ ... $\underline{2}$ ... $\underline{2}$ ... , sau **imbricate** - ... $\underline{1}$... $\underline{2}$... $\underline{2}$... $\underline{1}$... , dar **nu se pot „intersecta”**, ca în situaţia ... $\underline{1}$... $\underline{2}$... $\underline{1}$... $\underline{2}$... . Dacă o apariţie a unei variabile x se află în interiorul mai multor domenii imbricate, desigur că ea aparţine tuturor. Apariţiile diferite ale unui simbol pot avea semnificaţii (valori) diferite. Astfel, în ... $\underline{1}$... $\underline{2}$... $\underline{3}$ ... $x_{(1)}$ ... $\underline{3}$ ... $x_{(2)}$ ... $\underline{2}$... $\underline{1}$... , apariţia lui x notată $x_{(1)}$ şi cea notată $x_{(2)}$, pot practic fi considerate ca fiind nume de variabile diferite. În ceea ce priveşte *parantezele adăugate*, ele pot fi fie *restaurări* ale parantezelor şterse, fie ne folosim de (iv) (**Definiţia 3.1**, **Pas inductiv** extins).

Exemplu. Fie formula:

$F = Q(x) \vee (\exists x)(\forall y)(P(f(x), z) \wedge Q(a) \vee (\forall x)R(x, z, g(x)).$

Semnificaţia simbolurilor care apar:

- **x, y, z** $\in \mathcal{X}$. Avem x, y $\in \mathcal{X}$ deoarece sunt (şi) nume ale unor cuantificatori, iar z $\in \mathcal{X}$ conform convenţiilor făcute (e literă de la sfârşitul alfabetului). Variabila x apare de 4 ori în F (de la stânga la dreapta: x are o apariţie liberă în Q(x); o apariţie legată în P(f(x), z), ea fiind în domeniul sintactic al cuantorului $(\exists x)$; o altă apariţie legată, pe prima poziţie din R(x, z, g(x)), aici apariţia fiind şi în domeniul sintactic al cuantorului $(\forall x)$; o a treia apariţie legată, în acelaşi domeniu sintactic, este în R(x, z, g(x)), şi anume în g(x)). Variabila *y nu apare nici o dată în F, deşi apare cuantorul* $(\forall y)$ (ea apare însă în restvar(F)). Variabila z apare de 2 ori în F, ambele apariţii fiind libere (prima dată în P(f(x), z) şi a doua oară în R(x, z, g(x))). În consecinţă, free(F) = {x, z}, leg(F) = {x}, var(F) = {x,z} (sau chiar {x, y, z} dacă acceptăm ca restvar(F) să fie introdus în var(F)).

- a $\in \mathcal{F}_0$ conform convenţiilor.
- Q $\in \mathcal{P}_1$, P $\in \mathcal{P}_2$, R $\in \mathcal{P}_3$, din scrierea sintactică (nici convenţiile nu sunt contrazise).
- f, g $\in \mathcal{F}_1$, din scrierea sintactică (la fel, convenţiile „spun" acelaşi lucru).

Domeniile sintactice ale cuantorilor. Domeniul sintactic al cuantorului (∃x) este format din (subformula) (∀y)(P(f(x), z) ∧ Q(a) ∨ (∀x)R(x, z, g(x)). Apoi, cel al lui (∀y), din (P(f(x), z) ∧ Q(a) ∨ (∀x)R(x, z, g(x)). În sfârşit, domeniul lui (∀x) este R(x, z, g(x)). Ţinând cont şi de priorităţile impuse cuantorilor, *formula corectă*, *total parantetizată* (conform **Definiţiei 3.1**), este de fapt:

F = (Q(x) ∨ (∃x)((∀y)(((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x)))))).

Orice amănunte legate de sintaxă ar trebui studiate numai pe această formă, practic noi neavând definiţii formale pentru cazul unor formule din care lipsesc anumite paranteze.

Mulţimea subformulelor formulei date.

subf(F) = {Q(x), P(f(x), z), Q(a), R(x, z, g(x)), (∀x)(R(x, z, g(x))), (R(x, z, g(x))), (P(f(x), z) ∧ Q(a)), ((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x))), (∀y)(((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x)))), (((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x)))), (∃x)((∀y)(((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x))))), ((∀y)(((P(f(x), z) ∧ Q(a)) ∨ (∀x)(R(x, z, g(x))))), F}. Dintre acestea, unele sunt **formule atomice**:

Q(x), P(f(x), z), Q(a) şi R(x, z, g(x)). În plus, Q(a) este o **formulă de bază**. Să reamintim din nou că avem o definiţie formală pentru subf(F) doar în cazul în care n-am eliminat paranteze din F şi că *subformulă a lui F este orice subcuvânt al său care aparţine lui* **LP1**.

Termii (funcţionali) care apar în formulă. Ei sunt în ordine: x, f(x), z, a, g(x) (de fapt, termii x şi z au câte două apariţii, ei fiind şi variabile).

Mulţimea subformulelor închise ale lui F. Nu avem asemenea subformule. Totuşi, prin convenţia stabilită, am admis că formulele care nu conţin deloc variabile (adică formulele de bază) sunt formule închise. Singura asemenea subformulă în cazul nostru este deci Q(a).

Matricea formulei date. Ea este (am eliminat şi parantezele corespondente devenite inutile în urma ştergerii cuantificatorilor):

$F^* = (Q(x) \vee ((P(f(x), z) \wedge Q(a)) \vee R(x, z, g(x)))$.

Închiderea universală a formulei este:

$F' = (\forall z)((\forall x) (Q(x) \vee (\exists x)((\forall y)(((P(f(x), z) \wedge Q(a)) \vee$

$\vee (\forall x)(R(x, z, g(x)))))))$.

Similar se obţine **închiderea existenţială**.

Fie acum **substituţia** $s = [x/c]\bullet[z/g(b)]\bullet[y/x]$. Dacă o aplicăm formulei F (iniţiale, nu celei parantetizate complet) găsim succesiv:

$(F)s =$

$(Q(c) \vee (\exists x)(\forall y)(P(f(x),z) \wedge Q(a) \vee (\forall x)R(x, z, g(x)))[z/g(b)]\bullet[y/z] =$

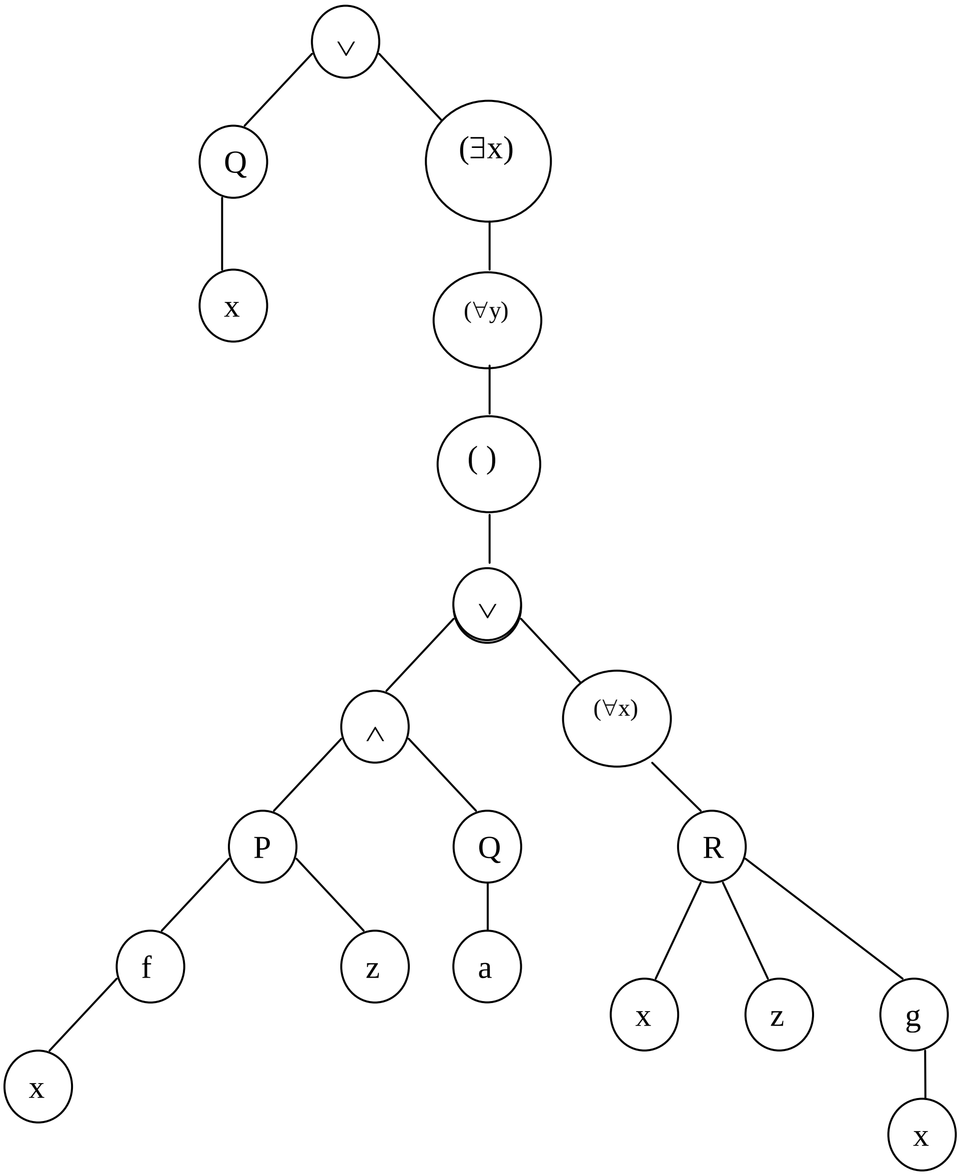
$= (Q(c) \vee (\exists x)(\forall y)(P(f(x), g(b)) \wedge Q(a) \vee (\forall x)R(x, g(b), g(x)))[y/x]$.

Ultima aplicare nu schimbă nimic deoarece y nu apare în F. Conform definiţiei rezultă că s **nu este permisă** pentru F (datorită faptului că în substituţia elementară [y/x], termul t = x conţine o variabilă care apare legată în F). Cum însă y este doar numele unui cuantificator, aplicarea lui s formulei F nu este dăunătoare în acest caz (o substituţie nepermisă generează de obicei grave anomalii semantice). Tot din acest motiv, s este şi **normalizată**.

Arborele ataşat formulei

$$Q(x) \vee (\exists x)(\forall y)(P(f(x), z) \wedge Q(a) \vee (\forall x)R(x, z, g(x)),$$

este (din nou, trebuia de fapt să considerăm formula corectă, total parantetizată; oricum, aceasta implică doar adăugarea unor noduri interne, etichetate cu „()”):



■

Putem acum introduce şi studia elementele de semantică privitoare la calculul cu predicate de ordinul I.

§2. Semantica logicii cu predicate de ordinul I

Înţelesul (semantica) unei formule F $\in$ **LP1** va fi, la fel ca în logica propoziţională, o valoare de adevăr 0, 1 $\in$ **B**, valoare obţinută într-un mod extensional. Elementul principal în definirea semanticii va rămâne noţiunea de *structură*. Deşi definiţia unei structuri şi găsirea unei valori de adevăr pentru o formulă va depinde practic doar de simbolurile care intervin în acea formulă, vom prefera să utilizăm în continuare să ne folosim de funcţiile totale în locul celor parţiale.

Definiţia 3.7. Se numeşte **structură** un cuplu $\mathcal{S} = <\mathcal{U}_\mathcal{S}, I_\mathcal{S}>$ în care $\mathcal{U}_\mathcal{S}$ este o mulţime **nevidă** numită **univers**, iar $I_\mathcal{S}$ este o funcţie (numită şi **interpretare**)

$$I_\mathcal{S} : \mathcal{X} \cup \mathcal{P} \cup \mathcal{F} \to \mathcal{U}_\mathcal{S} \cup [\mathcal{U}_\mathcal{S}^* \to \mathbf{B}] \cup [\mathcal{U}_\mathcal{S}^* \to \mathcal{U}_\mathcal{S}],$$

care satisface condiţiile:

- Dacă x $\in \mathcal{X}$, atunci $I_\mathcal{S}(\mathrm{x}) \in \mathcal{U}_\mathcal{S}$.
- Dacă P $\in \mathcal{P}_n$, atunci $I_\mathcal{S}(\mathrm{P}) : \mathcal{U}_\mathcal{S}^{\mathrm{n}} \to \mathbf{B}$.
- Dacă F $\in \mathcal{F}_\mathbf{n}$, atunci $I_\mathcal{S}(\mathrm{F}) : \mathcal{U}_\mathcal{S}^{\mathrm{n}} \to \mathcal{U}_\mathcal{S}$. ∎

În cele de mai sus, ca şi în restul materialului, [A $\to$ B] desemnează mulţimea tuturor funcţiilor totale având domeniul A şi codomeniul B, iar [A^* $\to$ B] denotă mulţimea tuturor funcţiilor de oricâte argumente (inclusiv 0) peste A, cu valori în B. Prin urmare, **interpretarea (semantica) unei variabile în structura $\mathcal{S}$ este un element din**

univers, interpretarea unui simbol predicativ n-ar este o funcţie de la $\mathcal{U}_S^n$ la {0, 1} (sau, uneori, *mulţimea elementelor din $\mathcal{U}_S{}^n$ pentru care valoarea în cauză este 1*), iar semantica unui simbol funcţional de aritate n este o funcţie de la $\mathcal{U}_S^n$ la $\mathcal{U}_S$. Pentru simplificarea exprimării, vom renunţa la indici dacă nu există confuzii şi **vom nota pe I_S tot cu $\mathcal{S}$**. Similar cu cazul logicii propoziţionale, orice structură va putea fi unic extinsă astfel încât să fie definită pentru toate elementele lui **LP1**.

Definiţia 3.8. Pentru fiecare structură $\mathcal{S} = <\mathcal{U}_S, I_S>$, vom numi **extensia sa imediată funcţia**

$\mathcal{S}'$: $\mathcal{X} \cup \mathcal{P} \cup \mathcal{F} \cup \mathcal{T} \cup$ **LP1** $\rightarrow \mathcal{U}_S \cup [\mathcal{U}_S^* \rightarrow \mathbf{B}] \cup [\mathcal{U}_S^* \rightarrow \mathcal{U}_S] \cup \mathbf{B}$,

dată constructiv în continuare. Pentru început, vom pune $\mathcal{S}'(a) = \mathcal{S}(a)$ (= $I_S(a)$), pentru fiecare $a \in \mathcal{X} \cup \mathcal{P} \cup \mathcal{F}$, ceea ce înseamnă că $\mathcal{S}'$ s-a definit, în particular, *pentru fiecare term elementar*. Fie acum orice $t \in \mathcal{T}$, adică orice $n \in \mathbf{N}^*$, orice $t_1, t_2, \ldots, t_n \in \mathcal{T}$ şi orice $f \in \mathcal{F}_n$, astfel încât $t = f(t_1, t_2, \ldots, t_n)$. Atunci $\mathcal{S}'(t) = \mathcal{S}(f)(\mathcal{S}'(t_1), \mathcal{S}'(t_2), \ldots, \mathcal{S}'(t_n))$ ($\in \mathcal{U}_S$). *Am încheiat astfel procesul de definire al lui $\mathcal{S}'$ pe $\mathcal{X} \cup \mathcal{P} \cup \mathcal{F} \cup \mathcal{T}$* şi rămâne să definim $\mathcal{S}'$ pe **LP1**. Vom face acest lucru în mod constructiv.

Baza. Fie $F = A \in \mathcal{A}t$. În această situaţie avem fie $A = P \in \mathcal{P}_0$ fie $A = P(t_1, t_2, \ldots, t_n)$, $n \in \mathbf{N}^*$, $t_1, t_2, \ldots\ldots, t_n \in \mathcal{T}$. În primul caz $\mathcal{S}'$ este deja definită ($\mathcal{S}'$(P) = $\mathcal{S}$(P) $\in$ **B**), iar în al doilea caz punem desigur $\mathcal{S}'$(P) = $\mathcal{S}$(P)($\mathcal{S}'(t_1)$, $\mathcal{S}'(t_2)$, ... , $\mathcal{S}'(t_n)$) $\in$ **B**.

Pas constructiv. Vom avea de considerat cazurile:

(i) F = ($\rceil$ F_1). Atunci $\mathcal{S}'$(F) = $\overline{\mathcal{S}'(F_1)}$.

(ii) F = ($F_1 \wedge F_2$). Atunci $\mathcal{S}'$(F) = $\mathcal{S}'(F_1) \bullet \mathcal{S}'(F_2)$.

(iii) F = ($F_1 \vee F_2$). Atunci $\mathcal{S}'$(F) = $\mathcal{S}'(F_1) + \mathcal{S}'(F_2)$.

(iv) F = ($\forall$x)(G). Atunci $\mathcal{S}'$(F) = 1 dacă şi numai dacă pentru fiecare u $\in$ $\mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}'_{[x/u]}$(G) = 1 unde $\mathcal{S}'_{[x/u]}$ este o interpretare care coincide în totalitate cu $\mathcal{S}'$ exceptând faptul că $\mathcal{S}'$(x) = u.

(v) F = ($\exists$x)(G). Atunci $\mathcal{S}'$(F) = 1 dacă şi numai dacă există (măcar) un element u $\in$ $\mathcal{U}_{\mathcal{S}}$ astfel încât $\mathcal{S}'_{[x/u]}$(G) = 1. ■

Vom pune, evident, şi $\mathcal{S}'$(F) = $\mathcal{S}'$((F)), pentru fiecare F $\in$ **LP1**. De asemenea, de acum înainte nu vom face nici o diferenţă între $I_{\mathcal{S}}$, $\mathcal{S}$, $\mathcal{S}'$, notând valoarea de adevăr a unei formule F $\in$ **LP1** într-o structură dată $\mathcal{S}$ prin $\mathcal{S}$(F) sau chiar $F^{\mathcal{S}}$ (de fapt, tehnic vorbind, am putea face acest lucru de-abia după demonstraţia **Teoremei 3.1**). În mod cu totul similar vor fi notate interpretările celorlalte simboluri în structura dată: $x^{\mathcal{S}}$, $c^{\mathcal{S}}$, $f^{\mathcal{S}}$, $P^{\mathcal{S}}$, $t^{\mathcal{S}}$, etc. Putem vorbi acum de noţiuni pe care le cunoaştem deja, cum ar fi *model*, *formule satisfiabile*, *valide*, *nesatisfiabile*, *consecinţă semantică*, *echivalenţă tare şi slabă*, etc. Aşa cum am mai precizat, dată o formulă F şi o structură $\mathcal{S}$, avem nevoie doar de valorile lui $\mathcal{S}$ pe simbolurile care apar în F, adică, la fel ca în cazul logicii propoziţionale, vom folosi titulatura de *structură corectă pentru o formulă (sau o mulţime de formule)*, pentru a denota restricţia unei structuri la simbolurile din $\mathcal{Alf}$ care apar într-o formulă (acesta fiind de fapt o funcţie parţială pe $\mathcal{X} \cup \mathcal{P} \cup \mathcal{F} \cup \mathcal{T} \cup$ **LP1**). Pot fi identificate şi anumite (clase de) structuri speciale atât la nivel sintactic cât şi la nivel semantic. *Sintactic*, să nu uităm că mulţimea de formule din calculul cu predicate de ordinul I depinde de alfabetul iniţial $\mathcal{Alf}$. Vom presupune astfel că mulţimea de variabile este numărabilă, că în mulţimea $\mathcal{P}$ cel puţin $\mathcal{P}_0$, $\mathcal{P}_1$, $\mathcal{P}_2$ sunt nevide, că mulţimea $\mathcal{F}$ conţine măcar un simbol funcţional (indiferent de aritate), etc. Fără restricţii de acest tip, există pericolul ca **LP1** să fie *trivială* (de exemplu, mulţimea vidă), *banală* (poate coincide cu **LP** sau chiar cu o subclasă a acesteia) sau *prea particulară* (putându-se exprima, de exemplu, doar relaţiile unare dintre diverse obiecte). *Semantic*, putem admite existenţa unor **simboluri speciale de predicate sau funcţii care să fie *interpretate identic în orice structură*.** Un exemplu este *relaţia de egalitate*, care poate fi considerată ca fiind reprezentată la nivel sintactic de un simbol predicativ de aritate 2 (să presupunem că $\mathcal{P}_2 \neq \emptyset$ şi „=” $\in \mathcal{P}_2$). În momentul în care considerăm o formulă F care conţine un asemenea simbol şi ne interesează existenţa unui model, *este normal să-l căutăm* ***doar*** *printre structuri de forma $\mathcal{S} = <\mathcal{U}_\mathcal{S}, I_\mathcal{S}>$ în care $=^\mathcal{S}(a, b)$ este egal cu* ***1*** *atunci şi numai atunci când a coincide cu b în universul dat* ($=^\mathcal{S} : \mathcal{U}_\mathcal{S} \times \mathcal{U}_\mathcal{S} \to$ **B**). Cu alte cuvinte, în asemenea situaţii, clasa structurilor/modelelor „admise” este restrânsă în mod forţat, ceea ce nu are întotdeauna efecte secundare favorabile. Astfel, în situaţia descrisă, clasa de formule este notată $\mathbf{LP1}_{\{=\}}$ (pe scurt, $\mathbf{LP1}_{=}$) şi se numeşte **calculul cu predicate de ordinul I *cu egalitate*** (punctăm iar: dacă $F \in \mathbf{LP1}_{=}$, atunci, dacă F conţine simbolul predicativ $= \in \mathcal{P}_2$, pentru a i se calcula valoarea de adevăr $\mathcal{S}(F)$ se iau în considerare doar structuri $\mathcal{S}$ în care „=” este interpretat *doar* ca fiind egalitatea pe universul respectiv). Acestea se vor numi **structuri/interpretări standard**. Similar se poate considera $\mathbf{LP1}_{\{+\}}$, $\mathbf{LP1}_{\{=,\ +\}}$, etc. Vom vedea că *problema satisfiabilităţii* pentru **LP1** (notată **SAT1**) este ***semidecidabilă***, *iar problema satisfiabilităţii* pentru $\mathbf{LP1}_{=}$ este ***nedecidabilă***. Un rol important în demonstraţii îl au *structurile Herbrand*.

Definiţia 3.9 (universuri şi **structuri Herbrand)**. Fie $F \in$ **LP1**. Se numeşte **univers Herbrand (ataşat lui F)**, mulţimea D(F) definită prin:

Baza. În D(F) se pun toate elementele din $\mathcal{F}_0$ care apar în F. Dacă F nu conţine nici o constantă, atunci se pune **forţat** în D(F) un element oarecare din $\mathcal{F}_0$ (numele rezervat standard, de obicei, este a).

Pas constructiv. Fie orice $n \in \mathbf{N}^*$, orice $f \in \mathcal{F}_n$ care apare în F şi termii oarecare $t_1, t_2, \ldots , t_n \in D(F)$. Atunci $f(t_1, t_2, \ldots , t_n) \in D(F)$.

O **structură Herbrand (pentru F)** este o structură (corectă pentru F) $\mathcal{H} = <\mathcal{U}_{\mathcal{H}}, I_{\mathcal{H}}>$, în care $\mathcal{U}_{\mathcal{H}} = D(F)$, iar $I_{\mathcal{H}}$ satisface condiţia că *interpretează fiecare term prin el însuşi*. Mai exact, $\mathcal{H}(f(t_1, t_2, \ldots , t_n)) =$

$f^{\mathcal{H}}(t_1^H, t_2^H, \ldots, t_n^H) = f(t_1^H, t_2^H, \ldots, t_n^H)$, pentru fiecare $n \in \mathbf{N}$ şi fiecare $t = f(t_1, t_2, \ldots, t_n) \in \mathcal{T}$. ■

Se poate spune că *D(F) este mulţimea termilor de bază construiţi cu simboluri din F*. Într-o structură Herbrand, dacă $f \in \mathcal{F}_0$ atunci $\mathcal{H}(f) = f$ şi în consecinţă dacă t este un term de bază avem şi $t^{\mathcal{H}} = t$. Interpretarea unei variabile este cea uzuală ($x^{\mathcal{H}} \in D(F)$ pentru fiecare $x \in \mathcal{X}$), la fel ca şi interpretarea simbolurilor predicative (ele vor fi funcţii oarecare peste D(F) cu valori în **B**). **A nu se confunda $f^{\mathcal{H}}(t_1^H, t_2^H, \ldots, t_n^H)$, care denotă aplicarea efectivă a funcţiei $f^{\mathcal{H}} : D(F)^n \to D(F)$ n-uplului $<t_1^H, t_2^H, \ldots, t_n^H>$, cu $f(t_1^H, t_2^H, \ldots, t_n^H)$ (valoarea aplicării anterioare), care este un term fără variabile aparţinând lui D(F), adică, în ultimă instanţă, un şir de caractere**. Dacă există o structură Herbrand care este model pentru o formulă F, atunci spunem şi că **F admite model Herbrand**.

Exemplu. Fie formula $F = (\forall x)(P(x, f(x)) \wedge Q(g(b, z))$.

(i) **Să se găsească o structură $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$, corectă pentru F, astfel încât $\mathcal{S}$ □ F**. Conform **Definiţiei 3.8**, vom avea $\mathcal{S}(F) = 1$ dacă şi numai dacă pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}_{[x/u]}(P(x, f(x)) \wedge Q(g(b, z)) = 1$, adică dacă şi numai dacă pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}_{[x/u]}((P(x, f(x))) = 1$ şi $\mathcal{S}_{[x/u]}(Q(g(b, z))) = 1$, adică dacă şi numai dacă pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $P^{\mathcal{S}[x/u]}(x^{\mathcal{S}[x/u]}, f^{\mathcal{S}[x/u]}(x^{\mathcal{S}[x/u]})) = 1$ şi $Q^{\mathcal{S}[x/u]}(g^{\mathcal{S}[x/u]}(b^{\mathcal{S}[x/u]}, z^{\mathcal{S}[x/u]})) = 1$, adică dacă şi numai dacă pentru fiecare u $\in \mathcal{U}_S$ avem $P^S(u, f^S(u)) = 1$ **şi** $Q^S(g^S(b^S, z^S)) = 1$. În acest moment, pentru obţinerea structurii cerute, se pune mai întâi problema alegerii unui univers $\mathcal{U}_S$ corespunzător. Ar trebui să ne gândim măcar la D(F) (a fost recent construit) şi la **N**, care este o mulţime cunoscută, având numeroase proprietăţi. Cum ambele mulţimi au acelaşi cardinal, vom încerca întâi cu $\mathcal{U}_S = \mathbf{N}$. Apoi, într-un prim pas ar trebui să găsim o relaţie $P^S : \mathbf{N} \times \mathbf{N} \to \mathbf{N}$ şi o funcţie $f^S : \mathbf{N} \to \mathbf{N}$, astfel încât pentru fiecare u $\in \mathcal{U}_S$ să avem $P^S(u, f^S(u)) = 1$. O relaţie binară pe **N**, cunoscută, este *relaţia de ordine parţială* $\leq$. Astfel, dacă alegem $P^S \cong \leq$, putem lua f^S ca fiind *funcţia succesor* ($f^S(n) = n + 1$, pentru fiecare n $\in$ **N**) şi atunci avem $P^S(u, f^S(u)) = 1$, pentru fiecare u $\in \mathcal{U}_S$. Structura deja sugerată mai trebuie completată de alegerea corespunzătoare a relaţiei $Q^S : \mathbf{N} \to \mathbf{N}$, a funcţiei $g^S : \mathbf{N} \times \mathbf{N} \to \mathbf{N}$ şi a valorilor $b^S \in \mathbf{N}$ şi $z^S \in \mathbf{N}$, astfel încât să avem şi $Q^S(g^S(b^S, z^S)) = 1$. Cum toate acestea sunt elemente precizate, există numeroase opţiuni convenabile. De exemplu, putem lua $b^S = 5$, $z^S = 10$, $g^S(m, n) = m + n$ pentru fiecare m, n $\in$ **N** (g^S este adunarea pe **N**) şi, în sfârşit, *$Q^S(n) = 1$ dacă şi numai dacă n este număr impar*. Trecerea la o structură Herbrand similară este uşoară, dacă ţinem cont de faptul că există o corespondenţă bijectivă între **N** (sau o submulţime finită a sa) şi D(F), oricare ar fi formula F.

(ii) **Să se găsească o structură $\mathcal{S} = <\mathcal{U}_S, I_S>$, corectă pentru F, astfel încât $\mathcal{S}$ □ F**. Procedăm identic cu cazul precedent, doar în final luăm de exemplu *$Q^S(n) = 1$ dacă şi numai dacă n este număr par*.

(iii) **Să se găsească universul Herbrand al lui F**. Conform **Definiţiei 3.9**, începem (**Baza**) prin a introduce în D(F) constantele care apar în F. Prin urmare, *iniţial* (conform convenţiilor deja adoptate), D(F) = {b}. Rămâne să aplicăm **Pasul constructiv** *cât timp este posibil*, „obţinând" D(F) = {b, f(b), f(f(b)), … } ∪ {g(b, b), g(g(b, b), b), ... } ∪ {f(g(b, b)), f(f(g(b, b))), …} ∪ {g(f(b), b), … } ∪ ... . Desigur că **D(F) este numărabilă**. Într-adevăr, considerând D(F) ca o mulţime de cuvinte peste alfabetul a = {b, g, f, (,), **,**}, putem defini *lungimea, l(w), unui cuvânt w ∈ a^* ca fiind egală cu numărul de caractere care îl compun (poate chiar exceptând parantezele)*. Atunci, pentru fiecare n ∈ **N**, mulţimea cuvintelor de lungime n va fi finită şi astfel D(F) va fi o reuniune numărabilă de mulţimi finite, deci numărabilă, adică aflată în corespondenţă bijectivă cu **N**. Mai mult, D(F) va fi chiar **recursiv enumerabilă** ([ŢIP]), adică *există un semialgoritm care, fără a avea nici o intrare, **listează** toate elementele din D(F)* (de exemplu, în ordinea crescătoare a lungimii acestora). În acest mod, putem presupune că a^* este total ordonată printr-o relaţie pe care o vom nota cu ∠ şi definită formal prin: *w_1 ∠ w_2 dacă şi numai dacă w_1 este înaintea lui w_2 în lista furnizată ca ieşire de semialgoritmul anterior* (putem chiar să menţinem în această listă o singură apariţie, de exemplu prima întâlnită în parcurgerea ei de la stânga la dreapta, a oricărui cuvânt). În consecinţă, mulţimea **D(F)** $\subseteq \mathcal{A}^*$ **este şi ea** ***total ordonată*** **prin restricţia relaţiei** $\angle$. **Vom prescurta acest lucru scriind D(F) = {w_1, w_2, ... }, ştiind şi că $w_1 \angle w_2 \angle \ldots$** . Să mai notăm faptul că D(F) este întotdeauna o mulţime *nevidă,* ea putând fi *finită în cazul în care F nu conţine simboluri funcţionale de aritate diferită de 0* (în restul cazurilor D(F) are aspectul de mai sus).

(iv) **Să se găsească o structură Herbrand $\mathcal{H}1 = <\mathcal{U}_{\mathcal{H}1}, I_{\mathcal{H}1}>$ care să fie model pentru F şi o alta $\mathcal{H}2 = <\mathcal{U}_{\mathcal{H}2}, I_{\mathcal{H}2}>$ care să nu fie model pentru F**. Ţinând cont de punctele anterioare şi de **Definiţia 3.9** (structurile Herbrand pentru o formulă F sunt întotdeauna corecte pentru F), avem D(F) = {w_1, w_2, ... } şi putem lua $P^{\mathcal{H}1} = P^{\mathcal{H}2} = \angle$, $f^{\mathcal{H}1}(w_i) = f^{\mathcal{H}2}(w_i) = w_{i+1}$ ($i \in$ N), $b^{\mathcal{H}1} = b^{\mathcal{H}2} = b \in$ D(F) (acest lucru este obligatoriu conform definiţiei unei structuri Herbrand), $z^{\mathcal{H}1} = z^{\mathcal{H}2} = b$ (de exemplu) şi $g^{\mathcal{H}1}(u, v) = g^{\mathcal{H}2}(u, v) = uv$ (aceasta însemnând concatenarea cuvintelor u şi v). În sfârşit, vom pune $Q^{\mathcal{H}1}(u) = 1$ dacă şi numai dacă $l(u)$ este număr par şi respectiv $Q^{\mathcal{H}2}(u) = 1$ dacă şi numai dacă $l(u)$ este număr impar. Rezultă imediat că $\mathcal{H}1$ □ F şi $\mathcal{H}2$ □ F. ■

Teorema 3.1 (de extensie). Pentru fiecare structură $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$, extensia sa imediată $\mathcal{S}'$ dată prin **Definiţia 3.8** este funcţie şi este unica funcţie având domeniul $\mathcal{X} \cup \mathcal{P} \cup \mathcal{F} \cup \mathcal{T} \cup$ **LP1** şi codomeniul $\mathcal{U}_{\mathcal{S}} \cup [\mathcal{U}_{\mathcal{S}}^* \to \mathbf{B}] \cup [\mathcal{U}_{\mathcal{S}}^* \to \mathcal{U}_{\mathcal{S}}] \cup \mathbf{B}$, care extinde $\mathcal{S}$ şi satisface condiţiile:

(i) $\mathcal{S}'((\neg F_1)) = \overline{\mathcal{S}'(F_1)}$, pentru fiecare F ∈ **LP1**.

(ii) $\mathcal{S}'((F_1 \wedge F_2)) = \mathcal{S}'(F_1) \bullet \mathcal{S}'(F_2)$, pentru fiecare F_1, F_2 ∈ **LP1**.

(iii) $\mathcal{S}'((F_1 \vee F_2)) = \mathcal{S}'(F_1) + \mathcal{S}'(F_2)$, pentru fiecare F_1, F_2 ∈ **LP1**.

(iv) $\mathcal{S}'((\forall x)(G)) = 1$ dacă şi numai dacă pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}'_{[x/u]}(G) = 1$ unde $\mathcal{S}'_{[x/u]}$ este o interpretare care coincide cu $\mathcal{S}'$ exceptând faptul că $\mathcal{S}'(x) = u$ (pentru fiecare $x \in \mathcal{X}$ şi fiecare G ∈ **LP1**).

Mai sus putem adăuga şi

(v) $\mathcal{S}'((\exists x)(G)) = 1$ dacă şi numai dacă există (măcar) un element $u \in \mathcal{U}_{\mathcal{S}}$ astfel încât $\mathcal{S}'_{[x/u]}(G) = 1$ (pentru fiecare $x \in \mathcal{X}$ şi fiecare G ∈ **LP1**).

Relaţia $\mathcal{S}'((F)) = \mathcal{S}'(F)$ o putem considera ca fiind adevărată chiar prin convenţie.

Demonstraţie. Faptul că $\mathcal{S}'(a) = \mathcal{S}(a)$ pentru fiecare $a \in \mathcal{X} \cup \mathcal{P} \cup \mathcal{F}$ ($\mathcal{S}'$ extinde $\mathcal{S}$) este adevărat din definiţie. Pentru a arăta faptul că $\mathcal{S}'$ este funcţie şi este unică având proprietăţile considerate, procedăm prin inducţie structurală (analog cu demonstraţia corespunzătoare de la logica propoziţională). Proprietatea de a fi funcţie a lui $\mathcal{S}'$ se păstrează succesiv pentru $\mathcal{T}$ şi **LP1** deoarece $\mathcal{S}$ este funcţie, unicitatea rezultând prin reducere la absurd, folosind direct relaţiile (i) – (v). ■

📖 ***Exerciţiul 3.3***. *Fie F o formulă din calculul cu predicate de ordinul I, cu sau fără egalitate,* $(\forall_*)F$ *închiderea sa universală şi* $(\exists_*)F$

închiderea sa existenţială. Ce se poate spune despre legătura dintre aceste formule în privinţa satisfiabilităţii (validităţii)?

În secţiunile/paragrafele următoare din **Capitolul 3** vom urma un drum oarecum similar cu calea parcursă în cazul **LP**, pentru a ajunge în situaţia să testăm satisfiabilitatea unei formule din **LP1** utilizând rezoluţia (adaptată pentru noul context). Avem nevoie de câteva rezultate preliminare.

§3. Forme normale în LP1

Să observăm mai întâi că **LP1** *poate fi considerată ca* ***o extensie reală*** *a lui* **LP***, atât* ***la nivel sintactic*** *cât şi* ***la nivel semantic***. Orice formulă de bază din **LP1** (formulă care nu conţine variabile) poate fi interpretată ca o formulă propoziţională (inclusiv elementele lui $\mathcal{P}_0$). Dacă formula în cauză nu conţine nici apariţii de conectori (şi, evident, nici de cuantori), ea va putea fi considerată drept o variabilă propoziţională. Conchidem că *din punct de vedere sintactic* avem într-adevăr **LP** $\subseteq$ **LP1** (însă este iar nevoie de nişte presupuneri suplimentare privind cardinalitatea unor mulţimi ca $\mathcal{P}_0$,$\mathcal{F}_0$, $\mathcal{F}_1$ sau $\mathcal{P}_1$). Mai mult, considerând orice structură $\mathcal{S}$ ca funcţie semantică în sensul **LP1**, ea poate fi considerată ca o extensie a funcţiei corespunzătoare aplicată pentru formulele care sunt (şi) elemente ale lui **LP** (iar această ultimă funcţie, după cum se constată direct din definiţii, reprezintă exact o structură în sensul **LP**). Vom vedea că relaţia dintre **LP** şi **LP1** este chiar mai profundă, **LP** *putând fi considerată ca o aproximare alui*

LP1, *la fel cum mulţimea numerelor raţionale este o aproximare a mulţimii numerelor reale*. Numele folosite, *logică propoziţională* respectiv *logică cu predicate de ordinul I*, sunt legate de *lipsa cuantificatorilor* respectiv de folosirea acestora pentru a *„lega” doar variabilele* (nu şi, de exemplu, simbolurile funcţionale sau cele predicative). Cuvântul *calcul*, utilizat ca un sinonim pentru *logică* în acest context, exprimă faptul că mulţimile (de formule) considerate nu sunt „amorfe”, ci pot fi „prelucrate” într-un mod „sistematic” (vorbim despre *calculul diferenţial şi integral*, de exemplu). O logică (calcul) în care ar fi permisă şi cuantificarea simbolurilor predicative (eventual şi a celor funcţionale), ar putea fi numită **logică (calcul) cu predicate de ordinul II** (**LP2**), dacă ţinem cont de faptul observat deja că (măcar) elementele lui $\mathcal{P}_0$ sunt formule atomice pentru **LP** dar „variabile” în **LP1** (analog, formulele atomice din **LP1** ar putea fi „variabile” în **LP2**). Fiecare nou tip de logică (**LP1**, **LP2**, …), trebuie să fie în acelaşi timp o extensie şi o aproximare a logicii anterioare (pe scurt, o **generalizare** sau o **logică de ordin superior**). După cum am mai subliniat de câteva ori, deşi rezultatele privind *complexitatea* sau *tratabilitatea (algoritmică a)* unor probleme importante sunt descurajatoare chiar pentru logici de ordin inferior, căutarea unor logici „mai generale” (de ordin superior, dar şi neclasice) este justificată prin *mărirea puterii de exprimare*.

Exemplu. Găsiţi o formulă F $\in$ **LP1**$_=$, care:

- Este satisfiabilă.

- Conţine un simbol funcţional binar $f \in \mathcal{F}_2$ şi un simbol constant $e \in \mathcal{F}_0$.
- Pentru fiecare $\mathcal{S}$ □ F avem: $<\mathcal{U}_{\mathcal{S}}, f^{\mathcal{S}}>$ este grup.

Am fi putut formula cele de mai sus astfel: *Să se găsească o formulă din calculul cu predicate de ordinul I cu egalitate care să exprime faptul că o anumită relaţie binară (numită f) peste o mulţime nevidă oarecare, determină pe aceasta o structură de grup (al cărui element neutru este notat e).* În acest caz, ar fi existat întrebări inevitabile, cum ar fi „Cum se poate defini *formal* faptul că ***o formulă exprimă*** o anumită cerinţă?” (acest lucru nu este posibil, deoarece măcar unul dintre elementele implicate nu poate avea o definiţie formală). Formularea adoptată mai sus are avantajul că este precisă, deoarece toţi termenii implicaţi au definiţii formale (formulă din **LP1**$_=$, formulă satisfiabilă, simbol funcţional binar, structură, grup, etc.). Pentru rezolvare va trebui să luăm în considerare şi diferitele clase de axiome (diferite) prin care se identifică un grup. Deoarece formula căutată F este din **LP1**$_=$ şi elementul neutru este precizat explicit, putem folosi axiomele „standard” (vom reveni la acest tip de problemă şi în **Capitolul 4**):

1. f este lege de compoziţie internă, binară, pe orice mulţime nevidă considerată:

F_1: $(\forall x)(\forall y)(\exists z)(f(x, y) = z) \wedge (\forall x)(\forall y)(\forall u)(\forall v)(f(x, y) = u \wedge f(x, y) = v \rightarrow u = v)$.

2. f este asociativă:

$$F_2: (\forall x)(\forall y)(\forall z)(f(x, f(y, z)) = f(f(x, y), z)).$$

3. f admite element neutru (notat e):

$$F_3: (\exists e)(\forall x)(f(x, e) = f(e, x) \wedge f(x, e) = x)).$$

4. Fiecare element x este simetrizabil în raport cu f:

$$F_4: (\forall x)(\exists x')(f(x, x') = f(x', x) \wedge f(x, x') = e)).$$

Cerinţele sunt astfel imediat satisfăcute dacă luăm

$F = F_1 \wedge F_2 \wedge F_3 \wedge F_4$. ■

Exerciţiul 3.4. *Considerând exemplul anterior, puteţi găsi o formulă F care să satisfacă aceleaşi cerinţe, exceptând folosirea explicită a lui e? Dar dacă am interzice şi folosirea explicită a simbolului de egalitate?*

Exerciţiul 3.5. *Să se găsească o formulă care să conţină un simbol predicativ binar P şi care să exprime faptul că P este o relaţie (binară) antisimetrică.*

Teorema 3.2 (de substituţie). Fie H ∈ **LP1**, oarecare. Fie orice F, G ∈ **LP1** astfel încît F este o subformulă a lui H şi G este tare echivalentă cu F. Fie H' formula obţinută din H prin înlocuirea (unei apariţii fixate a) lui F cu G. atunci H ≡ H'.

Demonstraţie. Se procedează prin inducţie structurală, într-un mod identic cu cazul **LP**. Situaţia în care sunt utilizaţi cuantificatorii trebuie tratat suplimentar. Reamintim astfel că afirmaţia de demonstrat este (∀H ∈ **LP1**)P(H), unde

P(H): (∀F, G, H' ∈ **LP1**)(((F ∈ subf(H)) şi

(H' se obţine din H înlocuind o apariţie fixată a lui F cu G) şi

$(F \equiv G)) \Rightarrow H \equiv H')$.

În **Pasul inductiv** singurul caz netratat anterior, în **Teorema 2.5** (pentru $H = (\exists x)H_1$ se procedează similar) este:

(iv) **$H = (\forall x)H_1$**. Fie orice $F \in subf(H) = \{H\} \cup subf(H_1)$. Dacă $F = H$, nu mai avem ce demonstra, la fel ca în cazul **Baza**. Să presupunem că $F \in subf(H_1)$ şi fie H_1' formula care se obţine din H_1 înlocuind o apariţie a lui F cu un G (nu uităm că $F \equiv G$). Ştiind, conform ipotezei inductive, că $H_1 \equiv H_1'$, faptul că $H' = (\forall x)H_1'$ este tare echivalentă cu H rezultă acum direct din definiţii. ■

Echivalenţele deja cunoscute (**Teorema 2.4**) pot fi completate cu altele, care se referă la cuantori.

Teorema 3.3. Pentru fiecare F, G $\in$ **LP1** şi fiecare x, y $\in \mathcal{X}$, sunt adevărate următoarele echivalenţe:

1. $\rceil(\forall x)F \equiv (\exists x)(\rceil F)$

   $\rceil(\exists x)F \equiv (\forall x)(\rceil F)$

2. Dacă x nu apare liber în G, atunci:

   $(\forall x)(F \wedge G) \equiv (\forall x)(F) \wedge G$

   $(\forall x)(F \vee G) \equiv (\forall x)(F) \vee G$

   $(\exists x)(F \wedge G) \equiv (\exists x)(F) \wedge G$

   $(\exists x)(F \vee G) \equiv (\exists x)(F) \vee G$

3. $(\forall x)(F) \wedge (\forall x)(G) \equiv (\forall x)(F \wedge G)$

   $(\exists x)(F) \vee (\exists x)(G) \equiv (\exists x)(F \vee G)$

4. $(\forall x)(\forall y)F \equiv (\forall y)(\forall x)F$

$(\exists x)(\exists y)F \equiv (\exists y)(\exists x)F$

5. Dacă x nu apare liber în F, atunci:

$(\forall x)F \equiv F$

$(\exists x)F \equiv F$.

Demonstraţie. Începem cu prima echivalenţă din 5. Fie astfel orice structură (corectă) $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$ astfel încât $\mathcal{S} \square (\forall x)F$. Aceasta înseamnă că $\mathcal{S}((\forall x)F) = 1$, adică pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}_{[x/u]}(F) = 1$. Dar $\mathcal{S}_{[x/u]}$ coincide cu $\mathcal{S}$, exceptând faptul că $\mathcal{S}_{[x/u]}(x) = u$ şi aceasta pentru fiecare apariţie (liberă) a lui x în F. Prin urmare, pentru că x nu are apariţii libere în F, $\mathcal{S}_{[x/u]}(F) = \mathcal{S}(F)$ (**q. e. d.**). A doua echivalenţă din 5. rezultă similar, iar 4. reprezintă o consecinţă imediată din definiţii. Mai demonstrăm doar prima echivalenţă din 2., restul lăsându-le pe seama cititorului. Fie din nou orice structură $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$ corectă pentru formulele care intervin. Avem succesiv:

$\mathcal{S}((\forall x)(F) \wedge G) = 1$ dacă şi numai dacă

pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$, $\mathcal{S}_{[x/u]}(F)$ şi $\mathcal{S}(G) = 1$ dacă şi numai dacă (folosim 5., pentru G)

pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$, $\mathcal{S}_{[x/u]}(F) = 1$ şi $\mathcal{S}_{[x/u]}(G) = 1$ dacă şi numai dacă

pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$, $\mathcal{S}_{[x/u]}(F\ G) = 1$ dacă şi numai dacă

$\mathcal{S}((x)(F \wedge G)) = 1$. ■

Observaţie. Ca o consecinţă a teoremei anterioare rezultă că sunt adevărate şi echivalenţele $(\forall x)(\forall x)F \equiv (\forall x)F$ şi $(\exists x)(\exists x)F \equiv (\exists x)F$ (chiar şi variantele $(\exists x)(\forall x)F \equiv (\forall x)F$, respectiv $(\forall x)(\exists x)F \equiv (\exists x)F$), care se pot generaliza pentru oricâte apariţii ale cuantorilor (F ar putea conţine la rândul ei cuantificatori). Se poate arăta însă că **nu sunt adevărate** echivalenţele (mai exact, există $x \in \mathcal{X}$, există F, G ∈ **LP1** astfel încât, mai jos, formulele din membrul stâng nu sunt echivalente cu cele corespunzătoare din membrul drept):

$(\forall x)F \vee (\forall x)G \equiv (\forall x)(F \vee G)$

$(\exists x)F \wedge (\exists x)G \equiv (\exists x)(F \wedge G)$

Să considerăm de exemplu prima relaţie. Chiar la nivel intuitiv cei doi membri nu exprimă acelaşi lucru deoarece faptul că F sau G este satisfăcută pentru o (aceeaşi) valoare a lui x nu înseamnă că fie F, fie G, sunt satisfăcute pentru (alte) valori ale lui x, eventual diferite. Astfel, putem lua drept contraexemplu:

- F = P(x).
- G = Q(x).
- $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$, $\mathcal{U}_{\mathcal{S}} = \mathbf{N}$, $P^{\mathcal{S}}(n) = 1$ dacă şi numai dacă n este număr par, $Q^{\mathcal{S}}(n) = 1$ dacă şi numai dacă n este număr impar (interpretarea lui x în $\mathcal{S}$ nu contează, toate apariţiile sale fiind în acest caz legate).

Nici echivalenţa $(\forall x)(\exists y)F \equiv (\exists y)(\forall x)F$ **nu este adevărată** pentru fiecare formulă F. ■

Putem trage şi concluzia că scrierea succesivă a unui aceluiaşi cuantor este inutilă, la fel fiind şi o nouă cuantificare a unei apariţii deja cuantificate (interpretarea unei variabile care nu are apariţii libere într-o formulă neavând efect în stabilirea valorii de adevăr a acelei formule, indiferent de structura considerată). Deoarece **echivalenţele anterioare pot fi interpretate şi ca *modalităţi de transformare* a formulelor care *păstrează validitatea* („**mutând cuantificatorii în faţă”**)**, putem considera că am făcut un prim pas spre obţinerea *formelor normale* pentru formulele din **LP1**. Deşi nu este imediat util, am preferat să începem cu rezultatul următor datorită importanţei sale în justificarea unor afirmaţii şi caracterului didactic al demonstraţiei. *Intuitiv, teorema afirmă că pentru a găsi semantica unei formule F (cu ajutorul unei structuri $\mathcal{S}$), formulă în care am substituit apariţiile libere ale unei variabile x cu un term de bază t, putem afla mai întâi semantica lui t în structura considerată şi apoi să găsim semantica lui F în noua structură* $\mathcal{S}_{[x/\mathcal{S}(t)]}$.

Teorema 3.4 (lema de translaţie). Fie F $\in$ **LP1**, x $\in \mathcal{X}$, t $\in \mathcal{T}$ un term de bază şi $\mathcal{S}$ orice structură corectă pentru formulele care apar. Atunci:

$$\mathcal{S}((F)[x/t]) = \mathcal{S}_{[x/\mathcal{S}(t)]}(F)$$

Demonstraţie. Arătăm prin inducţie structurală adevărul metaformulei $(\forall F)\mathsf{P}(F)$, unde:

$\mathsf{P}(F)$: (pentru orice x $\in \mathcal{X}$)(pentru orice t $\in \mathcal{T}$, term de bază)

(pentru orice structură $\mathcal{S}$)($\mathcal{S}((F)[x/t]) = \mathcal{S}_{[x/\mathcal{S}(t)]}(F)$).

Baza. F $\in \mathcal{A}t$.

(i) F = P $\in \mathcal{P}_0$. Neexistând variabile, afirmaţia este imediată.

(ii) Fie n $\in$ **N***, $t_1, t_2, \ldots, t_n \in \mathcal{T}$, P $\in \mathcal{P}_n$ astfel încât F = P$(t_1, t_2, \ldots, t_n)$.

Dacă x nu apare (liber) în F, afirmaţia este din nou evidentă. În caz contrar, este suficient să demonstrăm rezultatul pentru cazul în care există o (singură) apariţie (liberă) a lui x în F. Putem presupune, fără a restrânge generalitatea, că x apare la poziţia i (de la stânga la dreapta) în t_1. În acest caz, este clar că t_1 are forma $f(\ldots, \underset{i}{x}, \ldots)$, sau poate chiar coincide cu x. Atunci:

$\mathcal{S}((F)[x/t]) = \mathcal{S}((P(t_1, t_2, \ldots, t_n))[x/t]) =$

$= \mathcal{S}(P((t_1)[x/t], t_2, \ldots, t_n)) =$

$= \mathcal{S}((P(f(\ldots, x, \ldots)))[x/t], t_2, \ldots, t_n) =$

$= \mathcal{S}(P(f(\ldots, t, \ldots)), t_2, \ldots, t_n) =$

$= \mathcal{S}(P^{\mathcal{S}}(f^{\mathcal{S}}(\ldots, t^{\mathcal{S}}, \ldots)), \mathcal{S}(t_2), \ldots, \mathcal{S}(t_n)) =$

$= \mathcal{S}_{[x/S(t)]}$ (F), direct din definiţii, în contextul teoremei.

Pas inductiv.

(i) **F = (⌉ G)**. Atunci $\mathcal{S}((F)[x/t]) = \mathcal{S}((\rceil G)[x/t]) = \mathcal{S}(\rceil((G)[x/t])) = \mathcal{S}((G)[x/t])$.

Aplicând acum ipoteza inductivă pentru G, obţinem în continuare:

$\mathcal{S}((G)[x/t]) = \mathcal{S}_{[x/S(t)]}(G) = \mathcal{S}_{[x/S(t)]}(\rceil G) = \mathcal{S}_{[x/S(t)]}(F)$.

Să mai observăm că egalităţile anterioare sunt triviale dacă x nu are nici o apariţie liberă în F.

(ii) **F = (G ∘ H),** $\circ \in \{\wedge, \vee\}$. Se demonstrează analog cu cazul precedent, folosindu-se ipoteza inductivă pentru G şi H, regulile semantice corespunzătoare pentru $\vee$ şi $\wedge$, precum şi faptul că t nu conţine variabile (este term de bază).

(iii) **F = (∀y)G** (cazul **F = (∃y)G** poate fi demonstrat similar). Dacă y = x şi x nu apare liber în G, suntem într-un caz similar cu cel tratat în **Baza**. Presupunem că y ≠ x şi x apare liber în G (deci şi în F) măcar odată. Ca urmare, rezultă $\mathcal{S}$((F)[x/t]) = $\mathcal{S}$(((∀y)G)[x/t]) = $\mathcal{S}$((∀y)((G)[x/t])). Atunci:

$\mathcal{S}$((∀y)((G)[x/t])) = 1 dacă şi numai dacă pentru fiecare u ∈ $\mathcal{U}_{\mathcal{S}}$ avem

$\mathcal{S}_{[y/u]}$ ((G)[x/t])) = 1 dacă şi numai dacă (aplicăm ipoteza inductivă pentru G şi $\mathcal{S}_{[y/u]}$)

$\mathcal{S}_{[y/u][x/S(t)]}$ (G) = 1 dacă şi numai dacă ($t^{\mathcal{S}}$, u ∈ $\mathcal{U}_{\mathcal{S}}$, şi y ≠ x)

$\mathcal{S}_{[x/S(t)][y/u]}$ (G) = 1 dacă şi numai dacă

$\mathcal{S}_{[x/S(t)]}$ ((y)G) = 1 dacă şi numai dacă

$\mathcal{S}_{[x/S(t)]}$ (F) = 1. ■

Cu următorul rezultat se „începe construcţia” formelor normale.

Teorema 3.5 (lema de redenumire a variabilelor legate). Fie F = (○x)G , ○ ∈ {∀, ∃}, o formulă oarecare din **LP1**. Fie y o variabilă nouă (în sensul că ea nu apare în G). Atunci :

$$F \equiv (\circ y)(G[x/y]).$$

Demonstraţie. Imediată, prin inducţie structurală. ■

În cele de mai sus era suficient să presupunem că y, variabila nouă, nu apare liber în G.

Definiţia 3.10 (forma normala rectificată). O formulă F $\in$ **LP1** se numeşte *rectificată* (sau, se află în formă normală rectificată, pe scurt **FNR**) dacă nu conţine variabile care apar atât libere cât şi legate şi nu are cuantificatori care să lege aceeaşi variabilă, dar pe poziţii diferite în formulă (indiferent dacă este vorba de cuantificatori existenţiali sau universali). ■

Teorema 3.6. Pentru orice formulă din F $\in$ **LP1**, există o formulă rectificată F' $\in$ **LP1,** astfel încât F'≡ F.

Demonstraţie. Se aplică de un număr finit de ori **Lema de redenumire** formulei iniţiale F (implicit, se aplică şi **Teorema 3.3** şi **Teorema de substituţie**). ■

Definiţia 3.11 (forma normală prenex). O formulă F $\in$ **LP1** este în *formă normală prenex* (**FNP**, pe scurt) dacă $F = (\circ_1 y_1) \ldots (\circ_n y_n)G$, unde $n \in \mathbf{N}$, $\circ_i \in \{\exists, \forall\}$ $(i \in [n])$, iar $y_1, \ldots, y_n$ sunt toate variabilele distincte care apar (liber) în G. În plus, G nu mai conţine cuantificatori. ■

În cele de mai sus, cazul n = 0 se referă la absenţa cuantificatorilor ([0] = Ø).

Teorema 3.7. Pentru fiecare formulă F $\in$ **LP1**, există o formulă F' $\in$ **LP1**, care este în **FNP** şi este tare echivalentă cu F.

Demonstraţie. Fără a restrânge generalitatea, putem presupune că toate formulele care intervin sunt rectificate. Procedăm prin inducţie structurală, (meta)afirmaţia de demonstrat fiind evidentă.

Baza. F $\in$ $\mathcal{At}$. Aceasta este deja în **FNP**, având numărul de cuantificatori egal cu zero.

Pas inductiv.

(i) **F = ($\rceil$ F$_1$)**. Presupunem că F_1 este în **FNP** şi arătăm că F este în **FNP**. Conform ipotezei inductive pentru F_1, există G_1 aflată în **FNP** rectificată astfel încât $G_1 \equiv F_1$. Aceasta are aspectul $G_1 = (\circ_1 y_1) \ldots (\circ_n y_n)G$, unde $\circ_1, \circ_2, ..., \circ_n \in \{\exists, \forall\}$ şi G nu mai conţine cuantificatori. Atunci:

$F = (\rceil F_1) \equiv \rceil(\circ_1 y_1) \ldots (\circ_n y_n)G \equiv (\circ'_1 y_1) \ldots (\circ'_n y_n)(\rceil G)$, unde $\circ'_i = \exists$ dacă $\circ_i = \forall$ şi reciproc, pentru fiecare $i \in [n]$. Ultima formulă este în **FNP**.

(ii) **F = (F$_1$ o F$_2$), o $\in$ {$\wedge$, $\vee$}**. Conform ipotezei inductive, rezultă că există formulele

$G_1 = (\circ_1 x_1) \ldots (\circ_m x_m) G'_1$, $\circ_i \in \{\forall, \exists\}$, $i \in [m]$ şi

$G_2 = (\circ'_1 y_1) \ldots (\circ'_k y_k) G'_2$, $\circ'_i \in \{\forall, \exists\}$, $i \in [k]$, G_1 şi G_2 fiind în **FNPR** (*formă normală prenex rectificată*) şi echivalente respectiv cu F_1 şi F_2 . Atunci $F \equiv G_1 \circ G_2 = (\circ_1 y_1) \ldots (\circ_m y_m) G'_1 \circ$

$(\circ'_1 z_1) \ldots (\circ'_k z_k) G'_2 \equiv (\circ'_1 z_1) \ldots (\circ'_k z_k)(\circ_1 y_1) \ldots (\circ_m y_m)(G'_1 \circ G'_2)$.

În cele de mai sus, am aplicat de un număr finit de ori **Lema de redenumire**, **Teorema 3.3** şi **Teorema de substituţie**.

(iii) **F = (ox)G**, **o** $\in$ **{∀, ∃}**. Conform ipotezei inductive, există o formulă tare echivalentă cu G, aflată în **FNPR**, $G = (\circ_1 y_1) \ldots (\circ_n y_n)G'$, $\circ_i \in \{\forall, \exists\}$, $i \in [n]$. Dacă $x \in \{y_1, y_2, \ldots, y_n\}$, atunci aplicăm mai întâi **Lema de redenumire**. Să presupunem acum că $x \notin \{y_1, y_2, \ldots, y_n\}$. Atunci $F \equiv (\circ x)(\circ_1 y_1) \ldots (\circ_n y_n)G'$, formulă care se află în forma necesară. ■

Observaţie. Am arătat că pentru fiecare formulă din **LP1**, există o altă formulă din **LP1**, care este tare echivalentă cu ea şi care este în **FNP** rectificată (pe scurt, **FNPR**). Conform **Teoremei 3.3**, putem presupune şi că nu există în această formulă cuantificatori care să lege variabile care nu apar în ea şi nici cuantificatori (relativ la o aceeaşi variabilă) cu apariţii multiple. Cu alte cuvinte, din punctul de vedere al satisfiabilităţii, ne putem limita la studiul formulelor din **LP1** de forma $F = (\circ_1 y_1) \ldots (\circ_k y_k)F'$, unde $free(F') = \{y_1, y_2, \ldots, y_k\}$, iar F' este chiar matricea lui F, nemaiconţinând alţi cuantori ($\circ_1, \circ_2, \ldots, \circ_k \in \{\forall, \exists\}$). Prin urmare, folosind şi **Exerciţiul 3.3** (o formulă este satisfiabilă dacă şi numai dacă închiderea sa existenţială este satisfiabilă), putem spune că pentru testarea satisfiabilităţii unei formule din **LP1**, putem să ne limităm la clasa de formule având aspectul sintactic $F = (\circ_1 y_1)(\circ_2 y_2) \ldots (\circ_k y_k)F^*$, unde F^* este matricea lui F iar $leg(F) = var(F) = free(F^*) = \{y_1, y_2, \ldots y_k\}$. Prin urmare, această formulă este şi închisă, neconţinând apariţii libere de variabile. ■

Vom arăta în continuare că dacă există cuantificatori existenţiali în forma anterioară pentru o formulă, aceştia pot fi eliminaţi, fără a se afecta satisfiabilitatea. Testul de satisfiabilitate pentru formulele din **LP1** poate fi redus astfel la un test de satisfiabilitate pentru formule de forma $F = (\forall y_1) \ldots (\forall y_n)F^*$, unde $\{y_1, y_2, \ldots, y_k\} = free(F^*)$, F^* neconţinând cuantificatori.

Definiţia 3.12 (forma normală Skolem). O formulă $F \in$ **LP1** este în *formă normală Skolem* (**FNS**, pe scurt), dacă are aspectul $F = (\forall x_1) \ldots (\forall x_k)G$ unde G nu mai conţine cuantificatori (este matricea lui F), iar $x_1, x_2, \ldots, x_k$ sunt variabile distincte şi reprezintă exact variabilele care apar în G ($free(G) = \{x_1, x_2, \ldots, x_k\}$). F este în **formă normală Skolem clauzală** (**FNSC**, pe scurt), dacă este în **FNS** şi matricea sa este în **FNC** (forma normală conjuctivă) într-un sens similar cu **LP** (literalii reprezentând acum formule atomice din **LP1** sau negaţii ale lor). ■

Teorema 3.8. Pentru fiecare formulă F din **LP1**, există o altă formulă $F' \in$ **LP1**, care este în **FNSC** şi care este slab echivalentă cu ea.

Demonstraţie. Vom prezenta un algoritm prin care formula F' va fi construită efectiv din formula F.

Algoritm Skolem

Intrare: $F \in$ **LP1**. Fără a restrânge generalitatea, putem presupune că F este în **FNPR**, închisă.

Ieşire: F' $\in$ **LP1**, aflată în **FNS** (şi închisă), slab echivalentă cu F.

Metodă:

Pasul 1. F' : = F.

Pasul 2. **Cât_timp** (există cuantificatori existenţiali în F')

execută

2.1. Alege un asemenea cuantificator şi elimină-l.

2.2. Transformă formula F'.

Sf_Cât_timp.

Comentarii. Orice formulă intermediară prelucrată de algoritm are forma F' = $(\forall y_1) \ldots (\forall y_n)(\exists z)G$, unde G poate să conţină şi alţi cuantificatori (am pus în evidenţă primul cuantificator existenţial, alegerea fiind acum deterministă dacă ne gândim că parcurgem formula simbol cu simbol, de la stânga la dreapta). Atunci, în urma **Paşilor 2.1** şi **2.2**, F' va căpăta forma F' = $(\forall y_1) \ldots (\forall y_n)((G)[z/f(y_1, \ldots , y_n)])$ unde f este un simbol funcţional nou (în sensul că el nu mai apare în formulele considerate), f $\in \mathcal{F}_n$. Să notăm H $\cong (\forall y_1) \ldots (\forall y_k)(\exists z)G$, formula de tip F' existentă înainte de execuţia unui pas al algoritmului precedent şi cu H' = $(\forall y_1) \ldots (\forall y_k)(G)[z/f\,(y_1, y_2, \ldots y_k)]$ formula rezultată după execuţie. ■

Arătăm acum că *H este slab echivalentă cu H'*.

Presupunem întâi că H' este satisfiabilă. Există atunci o structură corectă $\mathcal{S} = <\mathcal{U}_\mathcal{S}, I_\mathcal{S}>$, astfel încât $\mathcal{S} \models$ H'. Ca urmare, pentru fiecare $u_1, u_2, \ldots , u_k \in \mathcal{U}_\mathcal{S}$ avem

$\mathcal{S}_{[y_1/u_1][y_2/u_2]...[y_k/u_k]}((G)[z/f(y_1, y_2, \ldots, y_k])) = 1$.

Pentru că variabilele $y_1, y_2, \ldots y_k$ sunt distincte şi apar liber în $(G)[z/f(y_1, y_2, \ldots, y_k)]$ şi pentru că structura $\mathcal{S}'$ (adică $\mathcal{S}_{[y_1/u_1][y_2/u_2]...[y_k/u_k]}$) coincide cu $\mathcal{S}$, exceptând valorile lui $y_1, y_2, \ldots y_k$ care sunt „puse" pe $u_1, u_2, \ldots, u_k$, semantica **LP1** ne permite să înlocuim în formula $G' = (G)[z/f(y_1, y_2, \ldots, y_k]$ pe $y_1, y_2, \ldots y_k$ respectiv cu nişte constante noi (nu apar nicăieri în formulele care intervin), distincte, $a_1, a_2, \ldots, a_k$ şi să extindem structurile $\mathcal{S}$ şi $\mathcal{S}'$ prin $a_i^{\mathcal{S}} = a_i^{\mathcal{S}'} = u_i$, pentru fiecare $i \in [k]$. În acest moment, interpretarea termului $t = f(y_1, y_2, \ldots, y_k)$ în structurile considerate va coincide cu cea a lui $f(a_1, a_2, \ldots, a_k)$. Putem astfel chiar înlocui pe t cu $f(a_1, a_2, \ldots, a_k)$, astfel încât el nu va conţine variabile. Avem deci $\mathcal{S}'(f(y_1, y_2, \ldots, y_k)) = \mathcal{S}'(f(a_1, a_2, \ldots, a_k)) = f^{\mathcal{S}'}(<\mathcal{S}'(a_1), \ldots, \mathcal{S}'(a_k)>) = f^{\mathcal{S}}(<u_1, \ldots, u_k>)$. Atunci $\mathcal{S}'((G)[z/t]) = 1$ şi t nu conţine variabile, astfel încât putem aplica **Lema de translaţie**, găsind $\mathcal{S}'_{[z/\mathcal{S}'(t)]}(G) = 1$, adică $\mathcal{S}'_{[z/\mathcal{S}(t)]}(G) = 1$. Sintetizând, putem spune că pentru fiecare $u_1, \ldots, u_k \in \mathcal{U}_{\mathcal{S}} = \mathcal{U}_{\mathcal{S}'}$, există $v \in \mathcal{U}_{\mathcal{S}}$ ($v = t^{\mathcal{S}} = f^{\mathcal{S}}(<u_1, \ldots, u_k>)$) astfel încât $\mathcal{S}'_{[z/v]}(G) = 1$. Prin urmare, $\mathcal{S}$ este model (şi) pentru H, adică H este satisfiabilă.

Invers, să **presupunem că H este satisfiabilă**. Fie orice $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$, structură corectă pentru H (atenţie, $\mathcal{S}$ nu este definită pentru simbolul f), astfel încât $\mathcal{S} \models$ H. Conform definiţiei, pentru fiecare $u_1, \ldots, u_k \in \mathcal{U}_{\mathcal{S}}$

există măcar un $v \in \mathcal{U}_{\mathcal{S}}$ astfel încât $\mathcal{S}_{[y_1/u_1][y_2/u_2]\ldots[y_k/u_k][z/v]}(G)=1$. Extindem pe $\mathcal{S}$ la $\mathcal{S}'$, corectă şi pentru H', punând (în rest, $\mathcal{S}'$ coincide cu $\mathcal{S}$): $f^{\mathcal{S}'}(<u_1, \ldots, u_k>) = v$. Prin urmare, pentru fiecare $u_1, \ldots, u_k \in \mathcal{U}_{\mathcal{S}} = \mathcal{U}_{\mathcal{S}'}$, avem $\mathcal{S}'_{[y_1/u_1][y_2/u_2]\ldots[y_k/u_k][z/f(u_1,\ldots,u_k)]}(G) = 1$. Luăm acum din nou termul t, $t = f(y_1, \ldots, y_k)$ şi folosind aceeaşi argumentaţie ca mai sus, putem aplica „invers" **Lema de translaţie**, obţinând $\mathcal{S}'_{[y_1/u_1][y_2/u_2]\ldots[y_k/u_k]}((G)[z/f(y_1,\ldots,y_k)]) = 1$, ceea ce înseamnă că $\mathcal{S}'$ satisface H', adică H' este satisfiabilă.

Prin urmare, orice execuţie a corpului buclei din **Algoritmul Skolem** elimină un cuantor existenţial, transformările făcute neschimbând caracterul formulei (de a fi sau nu satisfiabilă). Deci *fiecare* formulă intermediară de tipul F' este în **FNS** şi este slab echivalentă cu F. În sfârşit, pentru a aduce (F')* la forma clauzală, putem aplica orice algoritm cunoscut pentru cazul **LP** (renotând literalii pozitivi din F' prin nume de variabile propoziţionale distincte din $\mathcal{A}$). ■

Putem sintetiza rezultatele obţinute până în prezent în:

Teorema 3.9. Pentru fiecare formulă $F \in$ **LP1**, există o formulă $F' \in$ **LP1** astfel încât $F' \equiv_s F$, F' fiind în **FNSC** închisă ($F' = (\forall y_1) \ldots (\forall y_n)F^*$, $\{y_1, \ldots, y_n\} = free(F^*)$, F*, matricea lui F, fiind în formă normală conjunctivă). ■

Exemplul următor ne va sugera şi un algoritm pentru aflarea (unei) **FNSC** (de acum încolo vom presupune că orice formulă aflată în **FNSC** este şi închisă).

Exemplu. Fie

$F = (\forall x)(\exists y)(P(x, g(y), z)) \vee (\rceil(\forall x)(Q(x)) \wedge (\forall z)(\exists x)\rceil R(f(x, z), z)))$.

Să se găsească o formulă F', aflată în **FNSC**, slab echivalentă cu F.

Forma rectificată. Folosim **Lema de redenumire**, în principal (folosirea **Teoremei de substituţie** este implicită, iar echivalenţele exprimate prin **Teorema 3.3** trebuie utilizate imediat ce este posibil de a fi aplicate; *ordinea de aplicare a acestora nu este esenţială*). Găsim (prin redenumire z, redenumire x în două poziţii diferite şi aplicare **Teorema 3.3**, punctul 1.):

$F \equiv (\forall x)(\exists y)((P(x, g(y), z))) \vee ((\exists x)(\rceil Q(x)) \wedge (\forall z)(\exists x)\rceil R(f(x, z), z))\equiv$

$\equiv (\forall x)(\exists y)(P(x, g(y), z)) \vee ((\exists u)(\rceil Q(u)) \wedge (\forall t)(\exists v)\rceil R(f(v, t), t))$.

Forma normală prenex rectificată. $\wedge$ este prioritar faţă de $\vee$ (puteam deci să ne fi dispensat de nişte paranteze) şi u nu apare liber în $(\forall t)(\exists v)\rceil R(f(v, t), t))$. De aceea, $(\exists u)(\rceil Q(u)) \wedge (\forall t)(\exists v)\rceil R(f(v, t), t) \equiv (\exists u)(\rceil Q(u) \wedge (\forall t)(\exists v)\rceil R(f(v, t), t))$. În continuare, t şi v nu apar liber în $\rceil Q(u)$ şi atunci $(\exists u)(\rceil Q(u)) \wedge (\forall t)(\exists v)\rceil R(f(v, t), t)) \equiv (\exists u)(\forall t)(\exists v)(\rceil Q(u) \wedge \rceil R(f(v, t), t))$. Revenind, x şi y nu apar (liber) în $(\exists u)(\forall t)(\exists v)(\rceil Q(u) \wedge \rceil R(f(v, t), t))$ şi atunci $F \equiv (\forall x)(\exists y)(P(x, g(y), z) \vee (\exists u)(\forall t)(\exists v)(\rceil Q(u) \wedge \rceil R(f(v, t), t)))$. Acum u, t, v nu apar liber în $P(x, g(y), z)$ şi atunci:

$F \equiv (\forall x)(\exists y)(\exists u)(\forall t)(\exists v)(P(x, g(y), z) \vee \urcorner Q(u) \wedge \urcorner R(f(v, t), t))$.

Forma normală Skolem. Trebuie să eliminăm întâi cuantificatorii existenţiali, nu înainte de a porni cu o **formulă închisă**. Singura variabilă cu apariţii libere în ultima formulă este z şi obţinem imediat:

$F \equiv_s (\exists z)(\forall x)(\exists y)(\exists u)(\forall t)(\exists v)(P(x, g(y), z) \vee \urcorner Q(u) \wedge \urcorner R(f(v, t), t))$.

Aplicăm acum formulei din dreapta **Algoritmul Skolem**, adică efectuăm un număr de 4 execuţii a corpului buclei, în scopul de a elimina cei 4 cuantificatori existenţali.

I. $F \equiv_s (\forall x)(\exists y)(\exists u)(\forall t)(\exists v)(P(x, g(y), b) \vee \urcorner Q(u) \wedge \urcorner R(f(v, t), t))$, unde $b \in \mathcal{F}_0$.

II. $F \equiv_s (\forall x)(\exists u)(\forall t)(\exists v)(P(x, g(h(x)), b) \vee \urcorner Q(u) \wedge \urcorner R(f(v, t), t))$, unde $h \in \mathcal{F}_1$.

III. $F \equiv_s (\forall x)(\forall t)(\exists v)(P(x, g(h(x)), b) \vee \urcorner Q(i(x)) \wedge \urcorner R(f(v, t), t))$, unde $i \in \mathcal{F}_1$.

IV. $F \equiv_s (\forall x)(\forall t)(P(x, g(h(x)), b) \vee \urcorner Q(i(x)) \wedge \urcorner R(f(j(x, t), t), t))$, unde $j \in \mathcal{F}_2$.

Forma normală Skolem clauzală. Pornim cu:

$F^* = P(x, g(h(x)), b) \vee \urcorner Q(i(x)) \wedge \urcorner R(f(j(x, t), t), t)$.

Notăm $P(x, g(h(x)), b)$ cu A_1, $Q(i(x))$ cu A_2 şi $R(f(j(x, t), t), t)$ cu A_3. Atunci F^* devine $F^* = A_1 \vee \neg A_2 \wedge \neg A_3 = A_1 \vee (\neg A_2 \wedge \neg A_3) \equiv (A_1 \vee \neg A_2) \wedge (A_1 \vee \neg A_3)$. Prin urmare obţinem în final că formula noastră F este slab echivalentă (relaţiile $\equiv$ şi $\equiv_s$ sunt tranzitive; de asemenea, dacă $G \equiv H$ atunci $G \equiv_s H$) cu F', unde:

$$F' = (\forall x)(\forall t)((P(x, g(h(x)), b) \vee \neg Q(i(x))) \wedge$$
$$(P(x, g(h(x)), b) \vee \neg R(f(j(x, t), t), t))).\ \blacksquare$$

Existenţa formelor normale, precum şi a altor rezultate importante, ne permite un studiu sufient de detaliat al problemei satisfiabilităţii pentru calculul cu predicate de ordinul I, cu sau fără egalitate.

§4. Decidabilitate în LP1 ($LP1_=$)

Vom începe direct cu un rezultat cumva aşteptat.

Teorema 3.10. Fie F o formulă din calculul cu predicate de ordinul I fără egalitate, închisă şi aflată în **FNS**. Atunci F este satisfiabilă dacă şi numai dacă F admite un model Herbrand.

Demonstraţie.

Presupunem că F admite model Herbrand. Dacă F admite model Herbrand, ea este satisfiabilă prin definiţie.

Presupunem că F este satisfiabilă. Atunci există o structură $\mathcal{S} = <\mathcal{U}_\mathcal{S}, I_\mathcal{S}>$ corectă pentru formula dată astfel încât $\mathcal{S} \models F$. Arătăm că în acest caz F admite şi un model Herbrand, notat $\mathcal{H}_\mathcal{S} = <\mathcal{U}_\mathcal{H}, I_\mathcal{H}>$ (pe scurt, modelul va fi notat $\mathcal{H}$). Ţinînd cont de definiţia unei structuri Herbrand (orice term de bază este interpretat prin el însuşi), rămâne să interpretăm simbolurile predicative care intervin în formulă. Astfel, pentru fiecare n $\in$ **N**, pentru fiecare P $\in \mathcal{P}_n$ care apare în F şi pentru fiecare $t_1, t_2, \ldots, t_n \in \mathcal{U}_{\mathcal{H}}$ = D(F), vom pune:

(∗) $P^{\mathcal{H}}(<t_1, t_2, \ldots t_n>) = 1$ dacă şi numai dacă $P^{\mathcal{S}}(<t^{\mathcal{S}}_1, t^{\mathcal{S}}_2, \ldots t^{\mathcal{S}}_n>) = 1$.

Dacă F nu conţine constante, alegem o constantă arbitrară a $\in \mathcal{F}_0$, căreia îi dăm de asemenea o interpretare arbitară $a^{\mathcal{S}} \in \mathcal{U}_{\mathcal{S}}$. Pentru a arăta că $\mathcal{H} \models$ F, demonstrăm o **afirmaţie mai tare**:

Fie G orice formulă din **LP1** aflată în **FNP**, fără cuantori existenţiali, închisă şi conţinând simboluri funcţionale şi predicative care apar doar în F. Atunci, pentru fiecare structură $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, I_{\mathcal{S}}>$ corectă care este model pentru G, structura $\mathcal{H}$ construită ca mai sus este de asemenea model pentru G.

Demonstraţia afirmaţiei. Procedăm prin inducţie asupra numărului n de cuantificatori universali care apar în G.

Baza. n = 0. Atunci G nu conţine cuantificatori şi este închisă. Acest lucru înseamnă că G nu are variabile şi vom arăta că $\mathcal{S}$(G) = $\mathcal{H}$(G). Procedăm acum prin inducţie structurală asupra submulţimii lui **LP1** alcătuită din formulele de tipul G de mai sus.

Baza 1. G este formulă atomică fără variabile, deci G = P($t_1, t_2, \ldots, t_n$), n $\in$ **N**, P $\in \mathcal{P}_n$ şi apare în F, iar $t_1, t_2, \ldots, t_n \in \mathcal{U}_{\mathcal{H}}$. Egalitatea $\mathcal{S}$(G) = $\mathcal{H}$(G) urmează atunci direct din (∗).

Pas inductiv 1. Fie G_1, G_2 formule fără variabile (şi închise şi fără cuantificatori), având simbolurile funcţionale şi predicative din F, pentru care $\mathcal{S}(G_1) = \mathcal{H}(G_1)$ şi $\mathcal{S}(G_2) = \mathcal{H}(G_2)$. Trebuie să arătăm că avem $\mathcal{S}(G) = \mathcal{H}(G)$ pentru fiecare G de una din formele: $G = (G_1 \wedge G_2)$, $G = (G_1 \vee G_2)$ şi $G = (\neg G_1)$ (formula fiind închisă, nu mai trebuie considerate cazurile $(\forall x)(G_1)$ sau $(\exists x)(G_1)$). Şi acest lucru rezultă imediat din $(*)$ şi definiţia semanticii unei formule din **LP1**.

Pas inductiv. Să presupunem că afirmaţia noastră este adevărată pentru formule G de tipul considerat şi conţinând $n - 1$ cuantificatori. Fie acum G o formulă având cel mult n cuantificatori, închisă, aflată în **FNP**, neavând cuantori existenţiali şi fiind construită peste simboluri funcţionale şi predicative din F. Fie $\mathcal{S}$ o structură corectă pentru G, astfel încât $\mathcal{S} \models G$ şi $\mathcal{H}$ structura Herbrand dată prin $(*)$. Arătăm că $\mathcal{H} \models G$. Conform ipotezelor, $G = (\forall x)G'$, unde G' conţine cel mult $n - 1$ universali şi x apare liber în G' (în caz contrar, ajungem până la urmă din nou în situaţia din **Baza**). Astfel, deşi G' are restul proprietăţilor cerute pentru a putea aplica ipoteza inductivă, ea nu este formulă închisă. Dar, din $\mathcal{S}(G) = 1$ rezultă $\mathcal{S}((\forall x)G') = 1$, de unde găsim că pentru fiecare $u \in \mathcal{U}_{\mathcal{S}}$ avem $\mathcal{S}_{[x/u]}(G')=1$. Această relaţie este adevărată şi pentru acei u din $\mathcal{U}_{\mathcal{S}}$ care au proprietatea că sunt imagini prin structura $\mathcal{S}$ ai oricăror termi t fără variabile construiţi peste simboluri din F, $u = t^{\mathcal{S}} \in D(G) \subseteq D(F)$. Aplicând **Lema de translaţie** obţinem: $1 = \mathcal{S}_{[x/u]}(G') = \mathcal{S}_{[x/t]}(G') = \mathcal{S}((G')[x/t])$. În acest moment dispunem de formula G'([x/t]) care este închisă (t este term de bază), este în **FNP**, conţine cel mult n – 1 cuantificatori universali şi este construită peste simboluri din F. Putem în sfârşit aplica ipoteza inductivă, deoarece relaţia anterioară ne spune şi că $\mathcal{S}$ □ (G')[x/t]. Atunci $\mathcal{H}$ □ (G')[x/t] şi putem din nou aplica (în ultima relaţie) **Lema de translaţie** pentru a găsi $(\mathcal{H}_{\mathcal{S}})_{[x/t]}(G') = 1$ pentru fiecare t din D(G), adică $\mathcal{H}((\forall x)G') = 1$, ceea ce înseamnă de fapt că $\mathcal{H}(G) = 1$, adică $\mathcal{H}$ este model pentru G. Cu aceasta, afirmaţia este demonstrată (**q. e. d.**).

Pentru a încheia demonstraţia teoremei, aplicăm direct rezultatul precedent pentru G = F. ■

Observaţie. Tot ceea ce am afirmat sau demonstrat până la teorema anterioară este aplicabil şi în cazul calculului cu predicate de ordinul I cu egalitate. **Teorema 3.10** ***nu este însă adevărată şi pentru*** **LP1$_=$** , după cum se poate intui chiar urmărind demonstraţia, pentru succesul căreia relaţia (∗) este esenţială. Astfel, în cazul în care P este chiar egalitatea, ea trebuie interpretată standard în orice structură. Prin urmare, considerând b şi c două constante distincte, putem găsi uşor structuri $\mathcal{S}$ în care $b^{\mathcal{S}} = c^{\mathcal{S}}$, dar $b \neq c$ în **orice** structură Herbrand. Desigur că acest lucru nu înseamnă că n-am putea găsi o altă demonstraţie a rezultatului anterior, aplicabil pentru **LP1$_=$**. Din păcate, faptul nu este posibil, conform rezultatelor care urmează (nu toate demonstrate). ■

Teorema 3.11 (Löwenheim – Skolem). Fiecare formulă satisfiabilă din **LP1** admite model cel mult numărabil.

Demonstraţie. Se aplică direct **Teorema 3.10**. ■

Definiţia 3.13 (extensia Herbrand). Pentru fiecare formulă F închisă, aflată în **FNS,** $F = (\forall y_1) \ldots (\forall y_n)F^*$, $\{y_1, \ldots, y_n\} = free(F^*)$, F* fiind matricea lui F, extensia sa Herbrand este mulţimea

$E(F) = \{(F^*)[y_1/t_1]\bullet[y_2/t_2]\bullet \ldots \bullet[y_n/t_n] \mid t_1, t_2, \ldots, t_n \in D(F)\}$.

Dacă F este în **FNSC** (F are forma de mai sus, în plus F* fiind în **FNC**, $F^* = C_1 \wedge C_2 \wedge \ldots\ldots \wedge C_k$, $C_1, C_2, \ldots, C_k$ reprezentând clauze, adică literali din **LP1**), mulţimea

$$E'(F) = \bigcup_{i=1}^{k} E(C_i)$$

se numeşte **extensia Herbrand generalizată**. ■

Extensia Herbrand generalizată a unei formule este obţinută practic prin considerarea tuturor instanţelor clauzelor care compun matricea sa (formula fiind deja în **FNSC** şi considerată în reprezentarea cu mulţimi), instanţe obţinute prin aplicarea tuturor substituţiilor posibile cu termi de bază din D(F).

Teorema 3.12 (Church). Problema validităţii pentru logica cu predicate de ordinul I (fără egalitate) este nedecidabilă, dar semidecidabilă.

Demonstraţie. Conform, de exemplu, [MAS1]. ■

Observaţie. Ţinând cont de legătura dintre F şi ⌉F, se deduce imediat că problema **SAT1** este nedecidabilă, ca de altfel şi problema nesatisfiabilităţii (validităţii). Pe de altă parte, ***în cazul* LP1 *aceste probleme sunt totuşi semidecidabile***. ■

Teorema 3.13 (Gödel-Herbrand-Skolem). O formulă F ∈ **LP1** este satisfiabilă dacă şi numai dacă E(F) este satisfiabilă.

Demonstraţie. Fără a restrânge generalitatea, putem presupune că F este în **FNS** închisă. Conform **Teoremei 3.10**, F este satisfiabilă dacă şi numai dacă admite model Herbrand $\mathcal{H}$. Ca de obicei, putem considera că $F = (\forall y_1) \ldots (\forall y_n)F^*$ şi atunci, din $\mathcal{H}(F) = 1$ rezultă că pentru fiecare $t_1, t_2, \ldots, t_n \in D(F)$ avem $H_{[y_1/t_1][y_2/t_2]\ldots[y_n/t_n]}(F^*) = 1$. Deoarece termii în cauză sunt termi de bază, putem aplica **Lema de translaţie** de n ori, rezultând că $\mathcal{H}((F^*)[y_1/t_1]\bullet[y_2/t_2]\bullet\ldots\bullet[y_n/t_n]) = 1$, pentru fiecare $t_1, t_2, \ldots, t_n \in D(F)$. Prin urmare, am arătat că F este satisfiabilă dacă şi numai dacă $\mathcal{H} \models G$ pentru fiecare $G \in E(F)$ adică dacă şi numai dacă extensia Herbrand a lui F este satisfiabilă. ■

În **Teorema 3.13** putem lua E'(F) în loc de E(F) şi, mai mult, aşa cum de altfel am mai punctat, **elementele lui E'(F) pot fi privite drept clauze în sensul logicii propoziţionale**, deoarece literalii din **LP1** care le compun nu conţin variabile (astfel, interpretarea lor, 0 sau 1, în orice structură $\mathcal{S}$, depinde practic doar de interpretarea singurului simbol predicativ care apare, ceea ce se reduce la interpretarea unei variabile propoziţionale în sensul **LP**). Atunci are loc:

Teorema 3.14 (teorema lui Herbrand; teorema de compactitate pentru LP1). O formulă $F \in$ **LP1** este nesatisfiabilă dacă şi numai dacă există o submulţime finită a lui E'(F) care să fie nesatisfiabilă.

Demonstraţie. Direct din **Teorema 3.13** şi **Teorema de compactitate** pentru **LP**. ■

În acest moment putem spune că procedura următoare (intitulată chiar în întregime „**Procedura lui Gilmore**") poate fi *folosită pentru testarea nesatisfiabilităţii oricărei formule din **LP1***. **Pasul 1** este un algoritm în sine (a se vedea şi exemplul care urmează imediat **Teoremei 3.9**), formula găsită la sfârşitul execuţiei sale fiind slab echivalentă cu formula iniţială şi având aspectul $F = (\forall_*)F^*$, unde $F^* = C_1 \wedge C_2 \wedge \ldots \wedge C_k$. Extensia Herbrand generalizată E'(F) „rezultată" în urma aplicării **Pasului 2** trebuie interpretată ca fiind o listă de clauze din **LP**. Datorită faptului că E'(F) nu este recursivă ci doar recursiv enumerabilă, **Pasul 2** reprezintă doar un semialgoritm. După cum se observă, nici n-ar fi nevoie de obţinerea acestei liste dintr-o dată. Este nevoie doar de a putea selecta câte un nou element din ea „atunci când este necesar", conform **Pasului 3.3.2**, care ar putea fi reformulat prin *Obţine un nou element din E'(F)*. Practic, pornind de la ordinea pe D(F) deja sugerată (bazată pe lungimea termilor), se poate defini o ordine totală şi pe E'(F) (acest lucru nu ar implica decât o simplă extensie a unei relaţii de ordine definită pe o mulţime „suport” la un produs cartezian al acelei mulţimi cu ea însăşi, de oricâte ori). **Pasul 3**, şi numai el, este de fapt semialgoritmul cunoscut în literatura de specialitate ca *Procedura lui Gilmore* (*Procedura rezoluţiei de bază*). Aceasta nu se termină în cazul în care F este satisfiabilă şi conţine măcar un simbol funcţional de aritate cel puţin 1.

Semialgoritmul lui Gilmore

(Procedura rezoluţiei de bază)

Intrare: Orice formulă F $\in$ **LP1**.

Ieşire: „DA”, doar dacă F este nesatisfiabilă.

Metodă:

Pasul 1. Se transformă F într-o formulă aflată în **FNSC** (închisă), succesiv, prin rectificare (redenumire), găsirea **FNP** (**FNPR**), obţinerea închiderii existenţiale, obţinerea **FNS** şi apoi **FNSC**, formula rezultat notându-se, pentru simplitate, tot cu F.

Pasul 2. Se „obţine" $E'(F) = \{G_1, G_2, ... , G_n, ... \}$.

Pasul 3.

3.1. $M := \emptyset$.

3.2. $i := 0$.

3.3. **Repetă**

3.3.1. $i := i + 1$.

3.3.2. Alege $G_i \in E'(F)$.

3.3.3. $M := M \cup \{G_i\}$.

3.3.4. $M' := Res^*(M)$.

Până_când ($\square \in M'$).

Pasul 4. Tipăreşte „DA".

Trebuie însă să arătăm că (semi)algoritmul precedent „face ceea ce dorim". Să precizăm de la bun început că vom lua în considerare doar formule $F \in$ **LP1** pentru care $E'(F)$ este infinită. În caz contrar, rezultatele obţinute până în prezent ne „spun" că F poate fi privită ca o formulă din **LP**, nemaifiind necesară o tratare a acesteia în noul context.

Teorema 3.15. Procedura rezoluţiei de bază pentru **LP1** este corectă.

Demonstraţie. Trebuie de fapt să arătăm că pentru fiecare F $\in$ **LP1** având E'(F) infinită, semialgoritmul anterior se opreşte dacă şi numai dacă F este nesatisfiabilă. Acest lucru rezultă imediat din faptul că E'(F) este recursiv enumerabilă, din **Teorema lui Herbrand** şi din **Teorema rezoluţiei** pentru **LP**. ■

Putem acum justifica de ce ***logica propoziţională poate fi considerată ca o aproximare a logicii cu predicate de ordinul I***. Semnificaţia este aceea că **E(F) (sau E'(F)) aproximează pe F**, deoarece **testul de nesatisfiabilitate pentru F poate fi făcut testând de nesatisfiabilitate submulţimile finite ale lui E(F) (E'(F))**. Pentru detalii privind noţiunile de *număr cardinal şi ordinal*, *funcţie (mulţime) recursivă (recursiv enumerabilă)*, alte amănunte privind *calculabilitatea, complexitatea, tratabilitatea*, pot fi consultate: [ŢIP], [CAZ2], [COR], [AHO], [BÖR], [RIC], etc.

Exemplu. Fie formula

$F = (\forall x)(\forall y)((\rceil P(x) \vee \rceil P(f(c)) \vee Q(y)) \wedge P(y) \wedge (\rceil P(g(b, x)) \vee \rceil Q(b)))$.

Să se decidă dacă formula este nesatisfiabilă (satisfiabilă, validă), folosind **Procedura rezoluţiei de bază** (sau, altfel spus, folosind rezoluţia din calculul propoziţional). Să obesrvăm că F este deja în **FNSC**. Avem şi free(F^*) = {x, y}, iar $F^* = \{C_1, C_2, C_3\}$, unde:

$C_1 = \rceil P(x) \vee \rceil P(f(c)) \vee Q(y)$

$C_2 = P(y)$

$C_3 = \rceil P(g(b, x)) \vee \rceil Q(b)$, care la rândul lor se pot scrie ca mulţimi de literali din **LP1**. Putem trece astfel direct la **Pasul 2** al **Semialgoritmului lui Gilmore**, adică la aflarea mulţimii (listei) E'(F). O metodă generală amintită deja se bazează întâi pe enumerarea lui D(F), termii de bază fiind ordonaţi crescător, în ordinea lungimii lor (să precizăm şi că b precede pe c, f precede pe g, etc., şi abia apoi folosim indicaţia referitoare la lungime). Găsim, succesiv, ceva de genul:

$D(F) = \{b, c, f(b), f(f(b)), \ldots , f^{(n)}(b), \ldots , f(c), f(f(c)), \ldots , f^{(m)}(c), \ldots ,$
$\quad , g(b, b), g(b, c), g(c, b), g(c,c), g(b, f(b)), \ldots ,$
$\quad , g(b, g(b, b)), \ldots \}$

$E(C_1) = \{(C_1)[x/t_1]\bullet[y/t_2] \mid t_1, t_2 \in D(F)\} = \{\rceil P(b) \vee \rceil P(f(c)) \vee Q(b), \rceil P(b) \vee \rceil P(f(c)) \vee Q(c), \rceil P(c) \vee \rceil P(f(c)) \vee Q(b), \rceil P(c) \vee \rceil P(f(c)) \vee Q(c), \rceil P(b) \vee \rceil P(f(c)) \vee Q(f(b)), \ldots \} = \{\{\rceil P(b), \rceil P(f(c)), Q(b)\}, \{\rceil P(b), \rceil P(f(c)), Q(b)\}, \{\rceil P(b), \rceil P(f(c)), Q(b)\}, \ldots \}$

$E(C_2) = \{\{P(b)\}, \{P(c)\}, \{P(f(b))\}, \{P(f(f(b)))\}, \{P(f^{(n)}(b))\}, \ldots \}$

$E(C_3) = \{\rceil P(g(b, b)) \vee \rceil Q(b), \rceil P(g(b, c)) \vee \rceil Q(b), \rceil P(g(b, f(b))) \vee \rceil Q(b), \rceil P(g(b, f(f(b)))) \vee \rceil Q(b), \rceil P(g(b, f^{(n)}(b))) \vee \rceil Q(b), \ldots \} = \{\{\rceil P(g(b, b)), \rceil Q(b)\}, \{\rceil P(g(b, c)), \rceil Q(b)\}, \{\rceil P(g(b, f(b))), \rceil Q(b)\}, \{\rceil P(g(b, f(f(b)))) \vee \rceil Q(b)\}, \{\rceil P(g(b, f^{(n)}(b))), \rceil Q(b)\}, \ldots \}$

$E'(F) = E(C_1) \cup E(C_2) \cup E(C_3)$.

Aplicarea **Pasului 3** *ad literam* este foarte anevoioasă, deoarece ar implica (ar trebui să amintim şi de faptul că ar fi mai simplu să denotăm întâi formulele atomice de bază distincte de mai sus prin variabile propoziţionale distincte) un calcul succesiv nu numai de rezolvenţi ci chiar de Res-uri, care, cel puţin teoretic, sunt (alte) mulţimi infinite recursiv enumerabile. **Este însă suficient (pentru a trage concluzia că F este nesatisfiabilă) să găsim o demonstraţie prin rezoluţie a clauzei vide pornind cu o submulţime (finită) de clauze ale lui E'(F)**. Mai întâi găsim:

{⌉P(g(b, c)), ⌉Q(b)}   {P(g(b, c))}

⌉P(g(b, c))   P(g(b, c))

{⌉Q(b)}

Cele două clauze care intervin provin din C_3 (prin substituţia [x/c]) şi respectiv C_2 (prin substituţia [y/g(b, c)]). Conform definiţiei, termii respectivi sunt elemente din D(F). Acum continuăm cu:

{⌉P(f(c), ⌉P(f(c)), Q(b)}   {P(f(c)}

⌉P(f(c)   P(f(c)

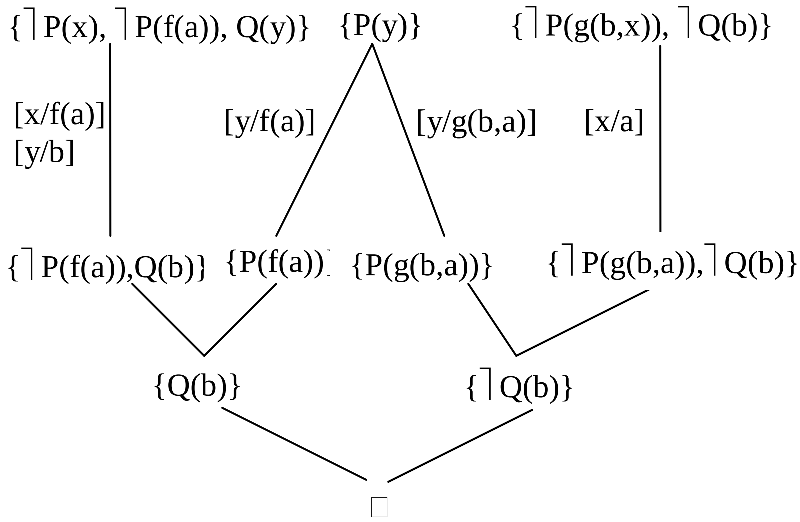
{Q(b)}

Deşi la ultimul pas este incorectă scrierea de două ori a elementului ⌉P(f(c), am făcut-o pentru a se identifica mai uşor clauza de provenienţă, care este C_1 (substituţia fiind [x/f(c)]•[y/b]). Pentru a doua clauză, ea este C_2, cu substituţia [y/f(c)]. În final, din {⌉Q(b)} şi {Q(b)} obţinem □. ■

Bazându-ne pe exemplul anterior, putem enunţa explicit rezultatul:

Teorema 3.16 (a rezoluţiei de bază). Fie F $\in$ **LP1** şi E'(F') extensia Herbrand generalizată a unei formule F', slab echivalente cu F şi aflată în **FNSC** (închisă). Atunci F este nesatisfiabilă dacă şi numai dacă există o demonstraţie prin rezoluţie (în sensul logicii propoziţionale) a lui □, pornind cu (o parte finită din) elementele lui E'(F') .
Demonstraţie. Imediată, din **Teorema lui Herbrand**, **Teorema 3.15** şi ţinând cont de legătura care există în **LP** între Res*(F) şi demonstraţiile prin rezoluţie. ■

Ca urmare, folosind **Teorema 3.16**, **Pasul 3** al **Procedurii rezoluţiei de bază** poate fi uşor modificat. Pentru simplificarea înţelegerii, îi prezentăm doar execuţia pentru cazul ultimului exemplu considerat:



Conchidem că aspectul **LP1** este următorul (subclasele indicate fiind nevide şi disjuncte):

<table>
<tr><td rowspan="2">Formule valide</td><td>Formule satisfiabile, nevalide, având doar model infinit</td><td rowspan="2">Formule nesatisfiabile</td></tr>
<tr><td>Formule satisfiabile, nevalide, având model finit</td></tr>
</table>

Prin urmare, oricâte eforturi am depune, *într-un limbaj suficient de bogat tot vor exista formule satisfiabile, nevalide şi având* ***doar*** *model infinit (numărabil), care nu va putea fi găsit algoritmic, într-un timp finit*. Totuşi, posibilitatea *utilizării unei rezoluţii proprii lui **LP1***, prin care să se evite măcar „scufundarea” în **LP** dacă nu şi aducerea formulelor la diferite forme normale, pare a fi o perspectivă benefică.

§5. Rezoluţie în LP1

Rezoluţia specifică (numită şi **pură**) pentru **LP1**, deşi diferită de rezoluţia de bază, păstrează ideea principală a rezoluţiei din **LP** şi anume că **la fiecare pas de rezoluţie se aleg două clauze şi se obţine o altă clauză (rezolvent), eliminând anumiţi literali prin „reducere" cu negaţiile lor**. Eliminările devin mai complicate, iar „esenţa" lor este (sub)procedura de **unificare**. A unifica două sau mai multe formule atomice din **LP1** înseamnă a găsi o substituţie pentru variabilele care intervin în acele formule (substituţia nefiind elementară sau de bază) astfel încât în urma aplicării substituţiei formulele atomice respective să coincidă (textual, ca şiruri de caractere). Obţinerea unui rezolvent nu va însemna numai simplă reducere a unui A cu un $\rceil$A, ci şi o unificare prealabilă a unor literali, prin care se „desemnează cine este A". Nu vom intra în prea multe detalii, pentru amănunte suplimentare putându-se consulta [MAS1], [CAZ1], etc.

Din motive legate de spaţiul tipografic, nu vom trata în această carte nici problematica legată de aplicarea rezoluţiei pure pentru cazul formulelor Horn şi nici situaţia rafinărilor rezoluţiei pure (**Capitolul 5** va conţine totuşi o scurtă introducere).

Exemplu. Vom relua formula din ultimul exemplu, despre care am arătat deja că este nesatisfiabilă utilizând rezoluţia de bază. Vom arăta că ea este nesatisfiabilă folosind rezoluţia pură din **LP1**. Să precizăm şi faptul că vom păstra toate notaţiile pe care le-am introdus pentru **LP**

(cum ar fi Res, Res*, etc.) deşi semnificaţiile exacte sunt de multe ori schimbate. După cum deja ştim, avem:

$F = (\forall x)(\forall y)((\rceil P(x) \vee \rceil P(f(c)) \vee Q(y)) \wedge P(y) \wedge (\rceil P(g(b, x)) \vee \rceil Q(b)))$,

$C_1 = \rceil P(x) \vee \rceil P(f(c)) \vee Q(y)$,

$C_2 = P(y)$,

$C_3 = \rceil P(g(b, x)) \vee \rceil Q(b)$,

$F^* = \{C_1, C_2, C_3\} =$

$\{\{\rceil P(x), \rceil P(f(c)), Q(y)\}, \{P(y)\}, \{\rceil P(g(b, x)), \rceil Q(b)\}\}$.

Considerăm pe rând următoarele cupluri de clauze:

- **C_1 şi C_2**. Din motive tehnice, nu trebuie să existe variabile comune în clauzele considerate în momentul în care încercăm să aplicăm un pas al rezoluţiei pure. De aceea facem mai întâi *substituţia de redenumire* [y/z] în C_2, găsind $C'_2 = \{P(z)\}$. Apoi, prin [x/z], *unificăm* mulţimea $\{\rceil P(x), \rceil P(z)\}$ (acest din urmă literal fiind complementarul celui conţinut de C'_2), găsind $Res(C_1, C'_2) = \{\rceil P(f(c)), Q(y)\} = C_4$.
- **C_4 şi C_2**. Facem aceeşi redenumire în C_2 şi lucrăm tot cu C'_2. Aplicând [z/f(c)], vom unifica mulţimea $\{\rceil P(f(c)), \rceil P(z)\}$, găsind $Res(C_4, C'_2) = \{Q(y)\} = C_5$.
- **C_3 şi C_2**. Nu mai avem nevoie de redenumiri, putând unifica $\{\rceil P(g(b, x)), P(y)\}$, prin [y/g(b, x)]. Găsim $Res(C_3, C_2) = \{\rceil Q(b))\} = C_6$.
- **C_5 şi C_6**. Nu avem nevoie de redenumiri, unificăm $\{Q(y), \rceil Q(b)\}$ prin [y/b] şi obţinem în final $Res(C_5, C_6) = \square$.

Astfel, am găsit o respingere utilizând rezoluţia pură, adică demonstraţia: C_1, C_2, C_4, C_5, C_3, C_6, □. ■

Deşi multe dintre rezultatele corespunzătoare pentru **LP** nu sunt adevărate în cazul **LP1** (dacă înlocuim rezoluţia **LP**, sau rezoluţia de bază pentru **LP1** cu rezoluţia pură), **Teorema rezoluţiei (pure)** are un enunţ similar cu **Teorema rezoluţiei pentru LP** şi în demonstraţia acesteia se foloseşte în mod esenţial **Teorema rezoluţiei de bază**. În acest mod, ***găsirea unei respingeri de tipul anterior este suficientă pentru a trage concluzia că formula în cauză este nesatisfiabilă.***

Nu putem încheia acest capitol fără a sublinia încă o dată faptul că **situaţia în $LP1_=$ este şi mai neplăcută**, cel puţin din punct de vedere teoretic. Se ştie astfel că, dacă limbajul iniţial este suficient de bogat, problema **SAT1** pentru **$LP1_=$** este chiar nedecidabilă „total”, adică nu este nici măcar semidecidabilă (conform **Teoremei lui Church**), ca în cazul **LP1**. Astfel, este suficient ca el să conţină ([MAS1]), înafara simbolului care reprezintă egalitatea, un simbol funcţional constant (de aritate 0), două simboluri funcţionale de aritate unu şi un simbol predicativ de aritate doi ($\wedge$, $\rceil$ şi $\exists$ sunt de asemenea necesari, ca şi existenţa în $\mathcal{X}$ a cel puţin două variabile distincte). Pentru ca **SAT1** pentru **$LP1_=$** să fie (măcar) semidecidabilă, este evident că ar trebui să putem arăta (măcar) că: **Orice formulă satisfiabilă din $LP1_=$ admite model cel mult numărabil**. Acest lucru nu poate fi, din păcate, adevărat, după cum se poate intui şi din enunţul teoremelor următoare (demonstraţiile pot fi găsite de asemenea în [MAS1]):

Teorema 3.17. Fie F $\in$ **LP1**, satisfiabilă, care admite un model finit $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, \mathcal{I}_{\mathcal{S}}>$ şi să presupunem că $|\ \mathcal{U}_{\mathcal{S}}\ | = n \in$ **N***. Atunci, pentru fiecare $m \geq n$, există un model $\mathcal{S}_m$ pentru F, cu cardinalul universului egal cu m. Mai mult, F are şi un model numărabil, care poate fi obţinut şi ca o „aproximare” a modelelor finite anterioare. ■

Teorema 3.18. Există o formulă închisă, satisfiabilă F $\in$ **LP1**, astfel încât pentru fiecare model $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, \mathcal{I}_{\mathcal{S}}>$ al lui F avem $|\ \mathcal{U}_{\mathcal{S}}\ | \geq 3$. ■

Teorema 3.19. Există o formulă satisfiabilă F $\in$ **LP1**$_=$, astfel încât pentru fiecare model $\mathcal{S} = <\mathcal{U}_{\mathcal{S}}, \mathcal{I}_{\mathcal{S}}>$ al lui F avem $|\ \mathcal{U}_{\mathcal{S}}\ | \leq 2$. ■

Inconvenientul practic al diferenţei dintre **LP1** şi **LP1**$_=$ poate fi reparat parţial (după cum am mai precizat, exemplificând chiar acest lucru în **Capitolul 5**) prin „ascunderea” simbolului „=” într-un simbol predicativ de aritate superioară.

§6. Recapitulare şi Index

Deşi teoria este foarte dezamăgitoare în privinţa rezultatelor pozitive (chiar dacă ne referim doar la calculul propoziţional, unde decidabilitatea **SAT** este „atenuată” de **netratabilitate**), necesitatea exprimării simple a situaţiilor reale a condus la acceptarea introducerii unor limbaje mai complexe (calculul cu predicate de ordinul I fără egalitate, calculul cu predicate de ordinul I cu egalitate, logicile cu predicate de ordinul II şi mai mare, etc.). Desigur că din punct de vedere formal rezultatele vor fi şi mai descurajante. **Există însă câteva argumente solide care fac ca utilizarea logicii să constituie într-adevăr o modalitate interesantă de a interpreta programarea** (vom analiza acest lucru şi în **Capitolul 5**). Pe parcursul **Capitolului 4**, am arătat că din punct de vedere formal atât **sintaxa** cât şi **semantica logicii cu predicate de ordinul I** constituie o **extensie** simplă şi naturală a **LP**, că logica propoziţională este chiar o **aproximare** a **LP1** (în cel mai pur sens matematic), că noţiunile deja folosite şi înţelese pentru **LP** pot fi extinse la **LP1** sau **LP1$_=$** (rezultatele generale fiind şi ele similare). Este vorba tot despre ***exprimarea realităţii prin formule atomice şi literali***, despre ***obţinerea de formule compuse din formule elementare*** (**variabilele, simbolurile funcţionale şi predicative** şi **cuantificatorii** fac diferenţa, dar la nivel intuitiv ele reprezintă chiar o necesitate de limbaj), despre ***forme normale*** şi despre ***posibilităţi efective de a testa satisfiabilitatea unei formule*** (de natură sintactică sau semantică). Problema **SAT1** pentru **LP1$_=$** este **nedecidabilă**, iar pentru **LP1** este **semidecidabilă**. Utilizarea rezoluţiei de bază ne arată o legătură mai profundă între **LP1** şi **LP**. Posibilitatea folosirii unui tip specific de rezoluţie - **rezoluţia pură -** pentru **LP1**, a unor clase particulare de formule (cum ar fi **clasa formulelor Horn**), **adaptarea strategiilor şi restricţiilor rezoluţiei** pentru cazul prezenţei variabilelor, **„scufundarea" realităţii în alte tipuri de logici**, rămân alternative viabile în **programarea logică**.

Pentru **Index**, vom aminti:

§7. Exerciţii

1. Definiţi constructiv free(F), $F \in$ **LP1**.
2. Determinaţi subf(F) pentru
   $F = (\forall x)(((P(x, g(a)) \wedge Q(z)) \vee R(u, f(v))))$.
3. Rezolvaţi **Exerciţiul 3.1**.
4. Rezolvaţi **Exerciţiul 3.2**.
5. Să se aplice substituţia: $s = [y/h(z)] \bullet [z/h(x)] \bullet [x/g(f(y))]$ formulei
   $F = (\forall x)(P(x, f(x)) \wedge Q(g(a, z))$.
6. Rezolvaţi **Exerciţiul 3.3**.
7. Fie formula $F = (\forall x)(\exists y)P(x, y, f(z))$. Să se decidă dacă formula este satisfiabilă, validă, sau contradicţie.
8. Găsiţi o formulă $F \in$ **LP1** care să conţină un simbol predicativ $P \in \mathcal{P}_2$ şi care să exprime faptul că P este o relaţie antisimetrică.

9. Găsiţi o formulă F $\in$ **LP1**$_=$ care să conţină un simbol predicativ P $\in$ $\mathcal{P}_2$ şi care să exprime faptul că P este o relaţie de echivalenţă.
10. Fie formula

    F = $(\exists x)(\exists y)(\exists z)(P(x, y) \wedge P(z, y) \wedge P(x, z) \wedge \rceil P(z, x))$ şi structurile $\mathcal{S} = <\mathcal{U}_\mathcal{S}, I_\mathcal{S}>$ date prin:

    - $\mathcal{U}_\mathcal{S}$ = **N**, $\mathbf{P}^\mathcal{S}$ = {<m, n> | m, n $\in$ N, m < n}.
    - $\mathcal{U}_\mathcal{S}$ = **N**, $\mathbf{P}^\mathcal{S}$ = {< m, m+1> | m $\in$ N}.
    - $\mathcal{U}_\mathcal{S}$ = $2^{\mathbf{N}}$, $\mathbf{P}^\mathcal{S}$ = {<A, B> | A, B $\subseteq$ N, A $\subseteq$ B}.

    Să se decidă care dintre aceste structuri sunt modele pentru F.
11. Arătaţi că pentru fiecare formulă din **LP1**:
    - F este validă dacă şi numai dacă închiderea sa universală este validă.
    - F este satisfiabilă dacă şi numai dacă închiderea sa existenţială este satisfiabilă.
12. Demonstraţi în detaliu **Teorema 3.1**.
13. Rezolvaţi **Exerciţiul 3.4**.
14. Rezolvaţi **Exerciţiul 3.5**.
15. Completaţi demonstraţia **Teoremei 3.2**.
16. Demonstraţi în detaliu **Teorema 3.5**.
17. Găsiţi o formulă F' din **LP1**, aflată în **FNSC** şi slab echivalentă cu formula:

    $F=(\forall x)(\exists y)((P(x,g(y), z) \vee \rceil(\forall x)Q(x)) \wedge (\forall z)(\exists x)\rceil R(f(x, z),z))$.
18. Arătaţi că în **LP1** există formule satisfiabile dar care nu admit nici un model finit.

19. Arătaţi că:

$$(\exists x)P(x) \rightarrow P(y) \equiv (\forall x)(P(x) \rightarrow P(y)).$$