BAZE DE CUNOŞTINŢE

Sinteza 2-2:

Modele şi consecinţă logică.

Nicolae Ţăndăreanu

Facultatea de Matematică-Informatică,

Universitatea din Craiova,

str.A.I.Cuza 13, 1100-Craiova, Romania

e-mail: ntand@oltenia.ro

1 Obiective

Obiectivele acestui capitol sunt următoarele:

- înţelegerea conceptelor de *validitate* şi *satisfiabilitate*
- înţelegerea conceptelor de *model* şi *model Herbrand*
- înţelegerea conceptului de *consecinţă logică*

2 Satisfiabilitate şi validitate.

Din punct de vedere formal, termenii şi formulele sunt stringuri care respectă anumite reguli de constituire. Ele sunt categorii abstracte dacă nu sunt luate în considerare în raport cu o *structură*. De îndată ce structura este fixată, prin intermediul funcţiilor *eval* şi *truth_val* aceste categorii abstracte pot fi *interpretate*. Prin funcţia *eval* termenul devine o entitate bine precizată a domeniului structurii. Prin funcţia *truth_val* formula capătă o *valoare de adevăr*. În cazul logicii bivalente această valoare este *true* sau $false$. O formulă dată poate căpăta valori de adevăr diferite prin interpretări diferite. După modul cum se comportă o formulă în raport cu o anumită structură sau în raport cu toate structurile posibile, formulele sunt de mai multe categorii. Această clasificare este prezentată în definiţia care urmează.

Definiţia 2.1 *O formulă φ se numeşte:*

- **satisfiabilă în structura** Σ *dacă există o asignare* $F \in D^{\mathcal{S}_V}$ *astfel încât*

$$truth\_val_{\Sigma}(\varphi, F) = true$$

notăm această proprietate cu $\varphi \in sat_{\Sigma}$

- **satisfiabilă** *dacă există o structură* Σ *astfel încât* $\varphi \in sat_{\Sigma}$*; altfel spus, dacă notăm* $sat = \bigcup_{\Sigma} sat_{\Sigma}$ *atunci proprietatea lui* φ *de a fi satisfiabilă se poate scrie sub forma* $\varphi \in sat$

- **validă în structura** Σ *dacă pentru orice asignare* $F \in D^{\mathcal{S}_V}$ *avem* $truth\_val_{\Sigma}(\varphi, F) = true$*; notăm această proprietate cu* $\varphi \in vd_{\Sigma}$

- **validă** *dacă pentru orice structură* Σ *avem* $\varphi \in vd_{\Sigma}$*; dacă notăm* $vd = \bigcap_{\Sigma} vd_{\Sigma}$ *atunci această proprietate se mai scrie sub forma* $\varphi \in vd$

Definiţia 2.2 *Fie* X *o mulţime de formule. O structură* Σ *se spune că este un* **model** *pentru* X *dacă* $X \subseteq vd_{\Sigma}$.

În particular X poate fi un program logic aparţinând uneia din cele patru categorii de programe prezentate în capitolul anterior.

Exemplul 2.1 *Considerăm următorul program logic disjunctiv pozitiv*

P:

$$\begin{cases} r(a) \vee r(b) \leftarrow \\ \\ p(f(x)) \vee q(x,x) \leftarrow r(x) \end{cases}$$

Observăm că:

- *baza acestui program este* $B = (\{a,b\}, \{f^{(1)}\}, \{p^{(1)}, q^{(2)}, r^{(1)}\})$
- *universul Herbrand este*

$$UH_B = \{a, b, f(a), f(b), \ldots, f^n(a), f^n(b), \ldots\}$$

- *baza Herbrand este* $BH_B = \{p(x), q(x,y), r(x)\}_{x,y \in UH_B}$

Considerăm structura $\Sigma = (D, k)$, *unde*

$D = \{0, 1, 2\}$, $k(a) = 2$, $k(b) = 0$,

$k(f) : D \longrightarrow D$, $k(f)(x) = 2 - x$,

$k(r) : D \longrightarrow \{true, false\}$, $k(r)(0) = true$,

$k(r)(1) = false$, $k(r)(2) = false$,

$k(p) : D \longrightarrow \{true, false\}$, $k(p)(x) = false$ *dacă şi numai dacă* $x = 1$,

$k(q) : D \times D \longrightarrow D$, $k(q)(x,y) = true$ *dacă şi numai dacă* $x \neq y$.

Să verificăm că Σ *este un model pentru* P. *Pentru aceasta să notăm cu* ψ_1, *respectiv* ψ_2 *cele două formule ale lui* P, *adică* $\psi_1 = r(a) \vee r(b)$ *şi* $\psi_2 = \forall x(p(f(x)) \vee q(x,x) \vee \neg r(x))$. *Observăm că* $\mathcal{S}_V = \{x\}$ *şi prin* *urmare trebuie să verificăm că pentru orice asignare* $F : \mathcal{S}_V \longrightarrow \{0, 1, 2\}$ *avem* $truth\_val_\Sigma(\psi_1, F) = true$ *şi* $truth\_val_\Sigma(\psi_2, F) = true$.

1) $truth\_val_\Sigma(r(a) \vee r(b), F) =$

$$truth\_val_\Sigma(r(a), F) \vee truth\_val_\Sigma(r(b), F) =$$

$$k(r)(k(a)) \vee k(r)(k(b)) = k(r)(2) \vee k(r)(0) = false \vee true = true$$

2) $truth\_val_\Sigma(\psi_2, F) = true$ *dacă şi numai dacă pentru orice* $c \in \{0, 1, 2\}$ *avem*

$$truth\_val_\Sigma(p(f(x)) \vee q(x, x) \vee \neg r(x), F\{x|c\}) = true$$

În consecinţă evem de analizat cazurile $c = 0$, $c = 1$ *şi* $c = 2$.

- *pentru* $c = 0$ *avem* $k(p)(k(f)(0)) \vee k(q)(0, 0) \vee \neg k(r)(0) = k(p)(2) \vee false \vee \neg true = true \vee false = true$
- *pentru* $c = 1$ *avem* $k(p)(k(f)(1)) \vee k(q)(1, 1) \vee \neg k(r)(1) = k(p)(1) \vee false \vee \neg false = true$
- *pentru* $c = 2$ *avem* $k(p)(k(f)(2)) \vee k(q)(2, 2) \vee \neg k(r)(2) = k(p)(0) \vee false \vee \neg false = true$

Să mai observăm că dacă luăm aceeaşi structură Σ, *dar în care modificăm pe* $k(b)$ *luând* $k(b) = 1$ *atunci structura obţinută nu mai este un model pentru programul* P *deoarece* $truth\_val_\Sigma(r(a) \vee r(b), F) = false$.

3 Consecinţa logică şi modele Herbrand.

La începutul acestei secţiuni definim conceptul de consecinţă logică şi studiem legătura dintre consistenţă şi acest concept.

Definiţia 3.1 *O mulţime X de formule se numeşte* **consistentă** *dacă admite cel puţin un model; în caz contrar mulţimea X se numeşte* **inconsistentă**.

Deseori întâlnim în cărţile de matematică afirmaţii de forma *definiţia T este consistentă.* Din punct de vedere intuitiv aceasta înseamnă că există o lume de obiecte astfel încât relaţiile stipulate în definiţie să fie adevărate. Ca exemplu, foarte simplu şi de foarte multe ori întâlnit, vom considera definiţia relaţiei de ordine parţială. Se ştie că o asemenea relaţie este caracterizată de următoarele formule:

- reflexivitatea: $\forall x p(x,x))$
- antisimetria: $\forall x \forall y[p(x,y) \wedge p(y,x) \rightarrow (x=y)]$
- tranzitivitatea: $\forall x \forall y \forall z[p(x,y) \wedge p(y,z) \rightarrow p(x,z)]$

Ce înseamnă că această definiţie este consistentă? Aceasta înseamnă că dacă notăm $X = \{\psi_1, \psi_2, \psi_3\}$, unde ψ_1, ψ_2 şi ψ_3 reprezintă formulele de mai sus, există o structură $\Sigma = (D,k)$ astfel încât $X \subseteq vd_\Sigma$. O asemenea structură este $\Sigma = (R,k)$, unde R este mulţimea numerelor reale, iar

$k(p) : R \times R \longrightarrow \{true, false\}$ este definită prin

$$k(p)(x, y) = \begin{cases} true \text{ dacă } x \leq y \\ \\ false \text{ altfel} \end{cases}$$

Structura considerată este un model pentru mulţimea X. Verificarea acestei proprietăţi se reduce la verificarea următoarelor relaţii, care nu arată alteceva decât faptul că relaţia $\leq$ pe mulţimea numerelor reale este o relaţie de ordine:

- pentru orice număr real c: $c \leq c$
- pentru orice numere reale c_1 şi c_2, dacă $c_1 \leq c_2$ şi $c_2 \leq c_1$ atunci $c_1 = c_2$
- pentru orice numere reale c_1, c_2 şi c_3, dacă $c_1 \leq c_2$, $c_2 \leq c_3$ atunci $c_1 \leq c_3$

Definiţia 3.2 *Fie X o mulţime de formule şi φ o formulă. Spunem că formula φ este o* **consecinţă logică** *a mulţimii X şi scriem $X \models \varphi$, dacă orice model pentru X este un model pentru φ.*

Teorema 3.1 *Fie X o mulţime de formule ground şi φ o formulă ground. Avem $X \models \varphi$ dacă şi numai dacă $X \cup \{\neg\varphi\}$ este inconsistentă.*

Proof. Presupunem că $X \models \varphi$. Prin absurd să presupunem că $X \cup \{\neg\varphi\}$ este consistentă şi fie Σ un model pentru $X \cup \{\neg\varphi\}$. În particular $\neg\varphi \in$

vd_Σ, deci $truth\_val_\Sigma(\neg\varphi, F) = true$ pentru orice $F \in D^{\mathcal{S}_V}$. Rezultă că $truth\_val_\Sigma(\varphi, F) = false$. Pe de altă parte Σ este un model pentru X şi deoarece $X \models \varphi$ rezultă că Σ este un model şi pentru φ. Aceasta înseamnă că $truth\_val_\Sigma(\varphi, F) = true$ pentru orice $F \in D^{\mathcal{S}_V}$, deci $\varphi \in vd_\Sigma$. Acest fapt contrazice proprietatea $truth\_val_\Sigma(\varphi, F) = false$.

Reciproc, să presupunem că $X \cup \{\neg\varphi\}$ este inconsistentă. Fie Σ un model pentru X. Avem $\neg\varphi \notin vd_\Sigma$ deoarece în cazul că $\neg\varphi \in vd_\Sigma$, Σ ar fi un model pentru $X \cup \{\neg\varphi\}$ ceea ce este imposibil. Există $F \in D^{\mathcal{S}_V}$ astfel încât $truth\_val_\Sigma(\neg\varphi, F) = \neg truth\_val_\Sigma(\varphi, F) = false$. Aşadar $truth\_val_\Sigma(\varphi, F) = true$ pentru cel puţin o asignare F. Ţinând seama de faptul că φ este formulă ground, rezultă că pentru orice $F \in D^{\mathcal{S}_V}$ avem $truth\_val_\Sigma(\varphi, F) = true$. Prin urmare $\varphi \in vd_\Sigma$. În consecinţă, $X \models \varphi$. ∎

Să aplicăm această teoremă pentru a demonstra că $\{\varphi, \psi\} \models \theta$, unde

$$\varphi = \forall x(p(x) \to q(x))$$

$$\psi = \forall x(q(x) \to r(x))$$

$$\theta = \forall x(p(x) \to r(x))$$

Aplicând teorema 3.1, trebuie să verificăm că mulţimea $\{\varphi, \psi, \neg\theta\}$ este inconsistentă. Prin absurd să presupunem că mulţimea este consistentă, deci există un model $\Sigma = (D, k)$ pentru ea. Altfel spus, $\{\varphi, \psi, \neg\theta\} \subseteq vd_\Sigma$, ceea ce înseamnă că pentru orice asignări F_1, F_2, F_3 avem:

$$truth\_val_\Sigma(\varphi, F_1) = true \tag{1}$$

$$truth\_val_\Sigma(\psi, F_2) = true \tag{2}$$

$$truth\_val_\Sigma(\neg\theta, F_3) = true \tag{3}$$

Din (1) şi (2) rezultă că pentru orice $c_1 \in D$ şi $c_2 \in D$ avem

$$truth\_val_\Sigma(q(x) \vee \neg p(x), F_1\{x|c_1\}) = true \tag{4}$$

$$truth\_val_\Sigma(r(x) \vee \neg q(x), F_2\{x|c_2\}) = true \tag{5}$$

Avem însă următoarele echivalenţe logice:

$$\neg\theta \approx \exists x \neg(r(x) \vee \neg p(x)) \approx \exists x (p(x) \wedge \neg r(x))$$

de unde rezultă că (3) are loc dacă şi numai dacă există $d \in D$ astfel încât

$$truth\_val_\Sigma(p(x) \wedge \neg r(x), F_3\{x|d\}) = true \tag{6}$$

Aplicând proprietăţile funcţiei de evaluare semantică, relaţiile (4), (5) şi (6) se mai scriu astfel:

$$k(q)(c_1) \vee \neg k(p)(c_1) = true \tag{7}$$

$$k(r)(c_2) \vee \neg k(q)(c_2) = true \tag{8}$$

$$k(p)(d) \wedge \neg k(r)(d) = true \tag{9}$$

Deoarece c_1 şi c_2 sunt elemente arbitrare în D, particularizăm aceste valori luând $c_1 = c_2 = d$. Din (9) rezultă $k(p)(d) = true$, $k(r)(d) = false$. Din (7) rezultă $k(q)(d) = true$, iar din (8) deducem $k(q)(d) = false$.

Această contradicţie ne arată că presupunerea făcută este falsă şi prin urmare mulţimea considerată este inconsistentă.

În continuare vom semnala o proprietate importantă a relaţiei de consecinţă logică. La prima vedere s-ar putea conchide că pentru orice formulă φ avem $X \models \varphi$ sau $X \models \neg\varphi$, ceea ce nu este adevărat. Pentru a da un contraexemplu vom considera formulele

$\varphi = p(a)$

$\psi = \forall x(p(x) \rightarrow p(f(x))$

$\theta = \forall x p(x)$

Ne propunem să arătăm că $\{\varphi, \psi\} \not\models \theta$ şi $\{\varphi, \psi\} \not\models \neg\theta$. Pentru aceasta vom arăta că mulţimile $\{\varphi, \psi, \neg\theta\}$ şi $\{\varphi, \psi, \theta\}$ sunt mulţimi consistente. Considerăm $D_1 = \{a, b, f(a), f(b), f(f(a)), f(f(b)), \ldots\}$ şi structura $\Sigma_1 = (D_1, k_1)$, unde

$k_1(f)(t) = f(t)$ pentru orice $t \in D_1$

$k_1(a) = a$

pentru orice $n \geq 0$ luăm $k_1(p)(f^n(a)) = true$ şi $k_1(p)(f^n(b)) = false$

Structura Σ_1 este un model pentru $\{\varphi, \psi, \neg\theta\}$ deoarece

- $truth\_val_{\Sigma_1}(\varphi, F) = truth\_val_{\Sigma_1}(p(a), F) =$

$$k_1(p)(k_1(a)) = k_1(p)(a) = true$$

- $truth\_val_{\Sigma_1}(\psi, F) = true$ dacă şi numai dacă pentru orice $c \in D_1$ avem $\neg k_1(p)(c) \vee k_1(p)(f(c)) = true$. Pentru a verifica această relaţie avem de analizat următoarele două cazuri:

  – dacă $c = f^s(a)$ atunci $k_1(p)(f(c)) = true$

  – dacă $c = f^s(b)$ atunci $k_1(p)(c) = false$, deci $\neg k_1(p)(c) = true$

  Observăm că în ambele cazuri relaţia dată este verificată.

- $truth\_val_{\Sigma_1}(\neg\theta, F) = true$ dacă şi numai dacă $truth\_val_{\Sigma_1}(\theta, F) = false$; ultima relaţie este adevărată dacă şi numai dacă există $c \in D_1$ astfel încât $k_1(p)(c) = false$; această proprietate este evident adevărată dacă luăm $c = f^s(b)$ pentru un s oarecare.

Dacă considerăm $\Sigma_2 = (D_2, k_2)$, unde

$$D_2 = \{a, f(a), f(f(a)), \ldots, f^s(a), \ldots\}$$

iar

- $k_2(f)(t) = f(t)$ pentru orice $t \in D_2$
- $k_2(a) = a$
- $k_2(p)(t) = true$ pentru orice $t \in D_2$

atunci constatăm că Σ_2 este un model pentru $\{\varphi, \psi, \theta\}$.

Prin rezultatul prezentat în **Teorema 3.1** vedem că problema demonstrării inconsistenţei unei mulţimi de formule închise este deosebit de importantă. Este dificil însă să arătăm că pentru o mulţime dată de formule nu există un model. În rezolvarea acestei probleme utilizăm conceptul de model Herbrand, introdus în definiţia care urmează.

Definiţia 3.3 *Fie B o bază. O* **interpretare Herbrand** *peste baza B este o submulţime nevidă $I \subseteq BH_B$. Dacă X este o mulţime de formule peste baza B atunci o interpretare Herbrand peste baza B o numim interpretare Herbrand a mulţimii X.*

Unei interpretări Herbrand I i se ataşează structura definită astfel:

Definiţia 3.4 *Hie B o bază şi I o interpretare Herbrand peste B. Definim structura $\Sigma_I = (UH_B, k_I)$, unde:*

i) pentru fiecare $f^{(n)} \in \mathcal{S}_F$, $k_I(f) : UH_B^n \longrightarrow UH_B$ definită prin $k_I(f)(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$

ii) pentru fiecare $p^{(n)} \in \mathcal{S}_P$, $k_I(p) : UH_B^n \longrightarrow \{true, false\}$ este definită astfel:

$$k_I(p)(t_1, \ldots, t_n) \begin{cases} true & p(t_1, \ldots, t_n) \in I \\ \\ false & altfel \end{cases}$$

Facem observaţia că în definiţia de mai sus numai $k_I(p)$ pentru $p \in \mathcal{P}_S$ depinde de I ($k_I(f)$ pentru $f \in \mathcal{S}_F$ nu depinde de I).

Considerăm că avem o submulţime $I \subseteq BH_B$ şi structura ataşată $\Sigma_I = (UH_B, k_I)$ definită mai sus. Putem uşor observa că:

- $eval_{\Sigma_I}(x, F) = F(x)$ pentru orice $x \in \mathcal{S}_V$ şi $F \in UH_B^{\mathcal{S}_V}$
- $eval_{\Sigma_I}(a, F) = k_I(a) = a$ pentru orice $a \in \mathcal{S}_C$
- $eval_{\Sigma_I}(f(a, b), F) = f(a, b)$ pentru $a, b \in \mathcal{S}_C$
- $eval_{\Sigma_I}(t, F) = t$ pentru orice $t \in UH_B$
- $truth\_val_{\Sigma_I}(p(t_1, \ldots, t_n), F) = k_I(p)(t_1, \ldots, t_n)=$

$$\begin{cases} true & \text{dacă } p(t_1, \ldots, t_n) \in I \\ \\ false & \text{altfel} \end{cases}$$

  pentru orice $p(t_1, \ldots, t_n) \in BH_B$

- pentru orice $x \in \mathcal{S}_V$ şi $p^{(1)} \in \mathcal{S}_P$:

$$truth\_val_{\Sigma_I}(p(x), F) = k_I(p)(F(x)) =$$

$$\begin{cases} true & \text{dacă} \quad p(F(x)) \in I \\ \\ false & \text{altfel} \end{cases}$$

şi în general pentru $p^{(n)} \in \mathcal{S}_P$ şi $x_1, \ldots, x_n \in \mathcal{S}_V$:

$$truth\_val_{\Sigma_I}(p(x_1, \ldots, x_n), F) =$$
$$\begin{cases} true & \text{dacă} \quad p(F(x_1), \ldots, F(x_n)) \in I \\ \\ false & \text{altfel} \end{cases}$$

- $truth\_val_{\Sigma_I}(\exists x p(x), F) = true$ dacă şi numai dacă există $c \in UH_B$ astfel încât $p(c) \in I$
- $truth\_val_{\Sigma_I}(\forall x p(x), F) = true$ dacă şi numai dacă pentru orice $c \in UH_B$ avem $p(c) \in I$

Definiţia 3.5 *O interpretare Herbrand I peste o bază B se numeşte* **model Herbrand** *pentru o mulţime X de formule peste B dacă perechea $\Sigma_I = (UH_B, k_I)$ este un model pentru X, adică $X \subseteq vd_{\Sigma_I}$.*

Teorema 3.2 *Fie P o mulţime de clauze. Există un model pentru P dacă şi numai dacă există un model Herbrand pentru P.*

Proof. Dacă există un model Herbrand pentru P atunci există un model pentru P. Rămâne să demonstrăm implicaţia de la stânga la dreapta. Presupunem că $\Sigma = (D, k)$ este un model pentru P. Considerăm interpretarea

$$I = \{p(t_1, \ldots, t_n) \in BH_P \mid p(t_1, \ldots, t_n) \in vd_\Sigma\}$$

structura $\Sigma_I = (UH_P, k_I)$ şi să arătăm că Σ_I este un model pentru P. Pentru simplitate vom considera că P are clauze de forma următoare:

$$p(t_1, \ldots, t_n) \leftarrow$$

$$\leftarrow q(x_1, \ldots, x_n)$$

$$A(x_1, \ldots, x_k) \leftarrow$$

$$A_1(x_1, \ldots, x_s) \vee A_2(y_1, \ldots, y_r) \leftarrow$$

$$A_1(x_1, \ldots, x_s) \vee A_2(y_1, \ldots, y_r) \leftarrow A_3(z_1, \ldots, z_q)$$

unde

$\{p, q, A, A_1, A_2, A_3\} \subseteq \mathcal{S}_P,\ t_1, \ldots, t_n \in UH_P$,

$\{x_1, \ldots, x_i, \ldots, y_1, \ldots, y_j, \ldots, z_1, \ldots, z_q\} \subseteq \mathcal{S}_V$

Ne propunem să verificăm că fiecare formulă considerată mai sus este în vd_{Σ_I}. Considerăm pe rând fiecare situaţie de mai sus:

1) $p(t_1, \ldots, p_n) \in BH_P \cap P \subseteq BH_P \cap vd_\Sigma = I$, deci

$$truth\_val_{\Sigma_I}(p(t_1, \ldots, t_n), F) = true$$

2) Notăm $\varphi = \forall x_1 \ldots \forall x_n \neg q(x_1, \ldots, x_n)$. Sunt echivalente următoarele afirmaţii:

- $truth\_val_{\Sigma_I}(\varphi, F) = true$

- $truth\_val_{\Sigma_I}(\neg q(x_1, \ldots, x_n), F\{x_1|c_1, \ldots, x_n|c_n\}) = true$ pentru orice $c_1, \ldots, c_n \in UH_P$
- $truth\_val_{\Sigma_I}(q(x_1, \ldots, x_n), F\{x_1|c_1, \ldots, x_n|c_n\}) = false$ pentru orice $c_1, \ldots, c_n \in UH_P$
- $q(c_1, \ldots, c_n) \notin I$ pentru orice $c_1, \ldots, c_n \in UH_P$

Deoarece $P \subseteq vd_\Sigma$, în particular $\varphi \in vd_\Sigma$ deoarece $\varphi \in P$. Rezultă $truth\_val_\Sigma(\varphi, G) = true$ pentru orice asignare G. Sunt de asemenea echivalente următoarele afirmaţii:

- $truth\_val_\Sigma(\varphi, G) = true$
- $truth\_val_\Sigma(\neg q(x_1, \ldots, x_n), G\{x_1|d_1, \ldots, x_n|d_n\}) = true$ pentru orice $d_1, \ldots, d_n \in D$
- $truth\_val_\Sigma(q(x_1, \ldots, x_n), G\{x_1|d_1, \ldots, x_n|d_n\}) = false$ pentru orice $d_1, \ldots, d_n \in D$
- $k(q)(d_1, \ldots, d_n) = false$ pentru orice $d_1, \ldots, d_n \in D$

Prin absurd să presupunem că $truth\_val_{\Sigma_I}(\varphi, F) = false$. Aceasta înseamnă că există $c_1, \ldots, c_n \in UH_P$ astfel încât $q(c_1, \ldots, c_n) \in I$, deci $q(c_1, \ldots, c_n) \in vd_\Sigma$. Echivalent, putem scrie că

$$truth\_val_\Sigma(q(c_1, \ldots, c_n), H) = true$$

pentru orice asignare H. Dar

$$truth\_val\Sigma(q(c_1, \ldots, c_n), H) = k(q)(eval_\Sigma(c_1, H), \ldots, eval_\Sigma(c_n, H))$$

Aşadar există $c_1, \ldots, c_n \in UH_P$ astfel încât

$$k(q)(eval_\Sigma(c_1, H), \ldots, eval_\Sigma(c_n, H)) = true$$

pentru orice asignare H. Să luăm o asignare H şi să notăm $e_1 = eval_\Sigma(c_1, H)$, ..., $e_n = eval_\Sigma(c_n, H)$. Evident elementele $e_1, \ldots, e_n$ sunt din D şi $k(q)(e_1, \ldots, e_n) = true$. Dar $k(q)(d_1, \ldots, d_n) = false$ pentru orice $d_1, \ldots, d_n \in D$. Contradicţia arată că presupunerea făcută nu este adevărată.

3) $\varphi = \forall x_1 \ldots \forall x_k A(x_1, \ldots, x_k)$. Ne propunem să verificăm că

$$truth\_val_{\Sigma_I}(A(x_1, \ldots, x_k), G\{x_1|d_1, \ldots, x_k|d_k\}) = true$$

pentru orice $d_1, \ldots, d_k \in UH_P$.

Echivalent trebuie să verificăm că $k_I(A)(d_1, \ldots, d_k) = true$ pentru orice $d_1, \ldots, d_k \in UH_P$. Dar $k_I(A)(d_1, \ldots, d_k) = true$ dacă şi numai dacă $A(d_1, \ldots, d_k) \in I = BH_P \cap vd_\Sigma$. Deoarece $A(d_1, \ldots, d_k) \in BH_P$, rămâne de verificat că $A(d_1, \ldots, d_k) \in vd_\Sigma$ sau echivalent, $k(A)(eval_\Sigma(d_1, H), \ldots, eval_\Sigma(d_k, H)) = true$ pentru orice asignare $H : \mathcal{S}_V \longrightarrow D$. Dar $eval_\Sigma(d_1, H), \ldots, eval_\Sigma(d_k, H)$ sunt elemente din D, iar $\varphi \in P \subseteq vd_\Sigma$. Din condiţia $\varphi \in vd_\Sigma$ rezultă că

$$truth\_val_\Sigma(\varphi, F) = true$$

pentru orice $F : \mathcal{S}_V \longrightarrow D$. Rezultă că

$$truth\_val_\Sigma(A(x_1, \ldots, x_k), F\{x_1|c_1, \ldots, x_k|c_k\}) = true$$

pentru orice $c_1, \ldots, c_k \in D$. Altfel spus, $k(A)(c_1, \ldots, c_k) = true$ pentru orice $c_1, \ldots, c_k \in D$. În particular avem

$$k(A)(eval_\Sigma(d_1, H), \ldots, eval_\Sigma(d_k, H)) = true$$

4) Celelalte două cazuri se verifică similar cu cazul anterior, cu observaţia că formula

$$A_1(x_1, \ldots, x_s) \vee A_2(y_1, \ldots, y_r) \leftarrow A_3(z_1, \ldots, z_q)$$

se mai scrie

$$A_1(x_1, \ldots, x_s) \vee A_2(y_1, \ldots, y_r) \vee \neg A_3(z_1, \ldots, z_q)$$

Lăsăm ca exerciţiu verificarea acestor cazuri. ■

Aşa cum se întâmplă în domeniul clasic al bazelor de date, un program logic poate fi *interogat.*

Definiţia 3.6 *Fie P un program logic. O* **interogare** *a lui P înseamnă:*

- *definirea unei formule*

$$\varphi = \exists x_1 \ldots \exists x_n (B_1 \wedge \ldots \wedge B_m)$$

  unde $B_1, \ldots, B_m$ sunt literali peste baza programului P, iar $x_1, \ldots, x_n$ sunt toate variabilele care apar în $B_1, \ldots, B_m$

- *stabilirea deducţiei $P \models \varphi$*

Formula φ se numeşte **interogare**.

Să observăm că $\neg\varphi \approx \forall x_1 \ldots \forall x_n(\neg B_1 \vee \ldots \vee B_m)$, deci $\neg\varphi$ este o clauză. Rămâne să vedem ce posibilităţi există pentru stabilirea deducţiei $P \models \varphi$. În acest sens avem următoarea proprietate, care arată că în cazul acestei deducţii ne putem restrânge numai la modelele Herbrand ale lui P.

Propoziţia 3.1 *Fie B o bază şi P un program logic peste B. Pentru orice întrebare φ peste B avem $P \models \varphi$ dacă şi numai dacă orice model Herbrand al lui P este model pentru φ*

Demonstraţie. Dacă $P \models \varphi$ atunci orice model al lui P este model pentru φ. În particular orice model Herbrand al lui P este model pentru φ. Următoarele trei afirmaţii sunt echivalente pe baza teoremelor 3.1 şi 3.2 precum şi a faptului că $P \cup \{\neg\varphi\}$ este o mulţime de clauze:

(a) $P \models \varphi$

(b) $P \cup \{\neg\varphi\}$ este inconsistentă

(c) nu există nici un model Herbrand pentru $P \cup \{\neg\varphi\}$

Să presupunem că orice model Herbrand al lui P este model pentru φ. Să demonstrăm că (c) este adevărată. Prin absurd presupunem că există un model Herbrand I pentru $P \cup \{\neg\varphi\}$. Aceasta înseamnă că $P \cup \{\neg\varphi\} \subseteq vd_{\Sigma_I}$. Conform presupunerii făcute, din $P \subseteq vd_{\Sigma_I}$ deducem $\varphi \in vd_{\Sigma_I}$. Aşadar $truth\_val_{\Sigma_I}(\varphi, F) = true$ pentru orice F, deci

$truth\_val_{\Sigma_I}(\neg\varphi, F) = false$ pentru orice F. Acest rezultat contrazice faptul că $\neg\varphi \in vd_{\Sigma_I}$. ∎

Se ştie că un program logic P este o mulţime finită de reguli şi fapte. Pentru orice întrebare φ, $\neg\varphi$ este un scop, deci $\neg\varphi \notin P$. Astfel, reuniunea $P \cup \{\neg\varphi\}$ nu este una trivială, ci înseamnă efectiv adăugarea clauzei $\neg\varphi$ la mulţimea de clauze din P. Pe de altă parte, observăm că dacă P este un program logic atunci BH_P este un model Herbrand pentru P. Această proprietatee se observă imediat dacă ţinem seama de faptul că orice instanţă a unei clauze din P are forma $A_1 \vee \ldots \vee A_k \vee \neg B_1 \vee \ldots \neg B_s$, iar $A_1, \ldots, A_k \in BH_P$. Mulţimea modelelor Herbrand ale unui program logic este nevidă. În această mulţime trebuie căutat cel mai mic model, deoarece acesta joacă un rol important în programarea logică.

Definiţia 3.7 *Un model Herbrand I al unui program logic P este:*

- **model minimal** *dacă nu există un model Herbrand I_1 pentru P astfel încât $I_1 \subset I$*
- **cel mai mic model** *dacă pentru orice model Herbrand J al lui P avem $I \subseteq J$*

Remarcă 3.1 *Teorema 3.2 nu este adevărată dacă P este o mulţime oarecare de formule. Într-adevăr, să considerăm că $P = \{\varphi\}$, unde $\varphi = p(a) \wedge \exists x \neg p(x)$. Evident, φ nu este o clauză, deci P nu este un program logic.*

Perechea $\Sigma_\varphi = \{D_\varphi, k_\varphi)$, *unde* $D_\varphi = \{0, 1\}$, $k_\varphi(a) = 0$, $k_\varphi(p)(0) = true$, $k_\varphi(p)(1) = false$ *este un model pentru* P. *Într-adevăr,*

$$truth\_val_{\Sigma_\varphi}(\varphi, F) =$$

$$truth\_val_{\Sigma_\varphi}(p(a), F) \wedge truth\_val_{\Sigma_\varphi}(\exists x \neg p(x), F) =$$

$$k_\varphi(p)(0) \wedge truth\_val_{\Sigma_\varphi}(\exists x \neg p(x), F)$$

Evaluând $truth\_val_{\Sigma_\varphi}(\exists x \neg p(x), F)$ *obţinem:*

$$truth\_val_{\Sigma_\varphi}(\exists x \neg p(x), F) = true$$

dacă şi numai dacă există $c \in D_\varphi$ *astfel încât* $truth\_val_{\Sigma_\varphi}(p(x), F\{x|c\}) = false$. *Deoarece* $k_\varphi(p)(1) = false$ *şi* $k_\varphi(p)(0) = true$, *rezultă că*

$$truth\_val_{\Sigma_\varphi}(\varphi, F) = true$$

Pentru mulţimea P *considerată avem* $B = (\{a\}, \{\}, \{p^{(1)}\})$, *deci* $BH_B = \{p(a)\}$. *Considerând o interpretare Herbrand* $I \subseteq BH_B$, *vedem că trebuie să luăm* $I = BH_B$. *Fie* $\Sigma_I = (UH_B, k_I)$ *structura asociată. Avem:*

- $truth\_val_{\Sigma_I}(p(a), F) = k_I(p)(k_I(a)) = k_I(p)(a) = true$ *pentru că* $p(a) \in I$

- $truth\_val_{\Sigma_I}(\exists x \neg p(x), F) = true$ *dacă şi numai dacă există* $c \in UH_B$ *astfel încât*

$$truth\_val_{\Sigma_I}(p(x), F\{x|c\}) = false$$

Deoarece $UH_B = \{a\}$ *şi* $k_I(p)(a) = true$, *rezultă că*

$$truth\_val_{\Sigma_I}(p(x), F\{x|a\}) = true$$

Aşadar $truth\_val_{\Sigma_I}(\exists x \neg p(x), F) = false$ *pentru orice* F. *În concluzie* P *nu admite nici un model Herbrand.*

4 TEME

TEMA 1

1) Consideraţi programul P:

$$\begin{cases} p(x) \vee q(g(x)) \leftarrow r(x) \\ r(a) \end{cases}$$

Arătaţi că P nu este inclus în vd.

2) Demonstraţi că φ este validă în Σ dacă şi numai dacă $\forall x \varphi$ este validă în Σ.

3) Demonstraţi că φ este validă dacă şi numai dacă $\neg\varphi$ nu este satisfiabilă.

TEMA 2

1) Fie φ o formulă care nu este închisă. Arătaţi că dacă $P \models \varphi$ atunci $P \cup \{\neg\varphi\}$ este inconsistentă. Arătaţi că reciproca nu este adevărată.

2) Definiţi formula $\varphi \equiv \psi$ prin $(\neg\varphi \vee \psi) \wedge (\varphi \vee \neg\psi)$. Arătaţi că $\varphi \equiv \psi \in vd_\Sigma$ dacă şi numai dacă $truth\_val_\Sigma(\varphi, F) = truth\_val_\Sigma(\psi, F)$ pentru orice $F \in D^{\mathcal{S}_V}$ (unde D este domeniul lui Σ).

3) Fie $p^{(1)} \in \mathcal{S}_P$ şi $c \in \mathcal{S}_C$. Fie formulele $\varphi_1 = ((\forall x)p(x) \rightarrow p(c)$ şi $\varphi_2 = (\exists x)[p(x) \rightarrow (\forall y)p(y)]$. Demonstraţi că $\{\varphi_1, \varphi_2\} \subseteq vd$.

TEMA 3

1) Fie $p^{(1)}, r^{(1)}$ simboluri de predicate. Arătaţi că următoarele formule sunt valide:

$$(\forall x p(x) \vee \forall x r(x)) \rightarrow \forall x(p(x) \vee r(x))$$

$$(\forall x p(x) \wedge \forall x r(x)) \leftrightarrow \forall x(p(x) \wedge r(x))$$

$$(\exists x p(x) \vee \exists x r(x)) \leftrightarrow \exists x(p(x) \vee r(x))$$

$$\exists x(p(x) \wedge r(x)) \rightarrow (\exists x p(x) \wedge \exists x r(x))$$

$$\forall x \neg p(x) \leftrightarrow \neg \exists x p(x)$$

$$\neg \forall x p(x) \leftrightarrow \exists x \neg p(x)$$

$$(\forall x(p(x) \rightarrow r(x)) \rightarrow (\forall x p(x) \rightarrow \forall x r(x))$$

$$(\exists x(p(x) \rightarrow r(x)) \leftrightarrow (\forall x p(x) \rightarrow \exists x r(x))$$

2) Fie $q^{(2)}$ un simbol de predicat. Arătaţi că următoarele formule sunt valide:

$$\forall x \forall y q(x,y) \leftrightarrow \forall y \forall x q(x,y)$$

$$\exists x \exists y q(x,y) \leftrightarrow \exists y \exists x q(x,y)$$

$\exists x \forall y q(x,y) \rightarrow \forall y \exists x q(x,y)$

TEMA 4

Fie I o interpretare Herbrand finită. Fie $\Sigma = \Sigma_I$ structura ataşată. Scrieţi un program în SWI Prolog care să calculeze valoarea funcţiei $truth\_val_{\Sigma}$.

TEMA 5

Fie $r^{(2)} \in \mathcal{S}_P$.

1) Demonstraţi că formula $(\exists y)(\forall x)r(x,y) \rightarrow (\forall x)(\exists y)r(x,y)$ este validă.

2) Studiaţi ce fel de formulă $(sat_{\Sigma}, sat, vd_{\Sigma}, vd)$ este formula

$$(\forall x)(\exists y)r(x,y) \rightarrow (\exists y)(\forall x)r(x,y)$$

TEMA 6

Considerăm programul logic P:

$$\begin{cases} p(x) \vee q(g(x)) \leftarrow r(x) \\ r(a) \leftarrow \end{cases}$$

a) Calculaţi universul Herbrand UH_P şi baza Herbrand HB_P a acestui program

b) Fie $I = \{p(a), r(g(a))\} \subseteq HB_P$. Definiţi interpretarea Herbrand Σ_I . Arătaţi că I nu este un model Herbrand pentru P.

c) Definiţi un model Herbrand pentru P.

TEMA 7

Considerăm programul P:

$$\begin{cases} p(x) \vee q(g(x)) \leftarrow r(x) \\ r(a) \leftarrow \end{cases}$$

1) Arătaţi că $\Sigma = (\{0,1\}, k)$ unde

$$k(a) = 1$$

$$k(g)(x) = \begin{cases} 0 \ dac\breve{a} \ x = 1 \\ 1 \ dac\breve{a} \ x = 0 \end{cases}$$

$$k(p)(x) = k(q)(x) = k(r)(x) = \begin{cases} false \ dac\breve{a} \ x = 0 \\ true \ dac\breve{a} \ x = 1 \end{cases}$$

este un model Herbrand pentru P.

2) Se ştie că dacă o mulţime S de clauze are un model atunci S are un model Herbrand. Să se aplice construcţia respectivă pentru a găsi un model Herbrand pentru P.

TEMA 8

Fie programul P de mai jos:

$$\begin{cases} p(f(x)) \leftarrow q(x) \\ p(x) \vee q(x) \leftarrow \\ p(a) \leftarrow \\ q(b) \leftarrow \end{cases}$$

1) Găsiţi un model Σ pentru P.

2) Utilizaţi acest model pentru a găsi un model Herbrand pentru P.