Logică computațională
Curs 3

Lector dr. Pop Andreea-Diana

Logica propozițiilor

Propoziţiile logice sunt modele ale afirmaţiilor propoziţionale care sunt fie *adevărate*, fie *false*.

Sintaxa logicii propoziţiilor

- alfabetul
  - $\Sigma_P = Var\_propoz \cup Conective \cup \{ \, (\, , \,) \, \}$
  - $Var\_propoz = \{ p, q, r, p_1, p_2, \ldots \}$
  - $Conective = \{ \neg , \wedge , \vee , \rightarrow , \leftrightarrow \}$

prioritate

$\neg$ negația
$\wedge$ conjuncția
$\vee$ disjuncția
$\rightarrow$ implicația
$\leftrightarrow$ echivalența

- regulile de formare a Formulelor propoziționale
  - F_P = mulţimea formulelor propoziţionale corect construite

  = cea mai mică mulțime de formule ce se poate construi cu regulile:
    - *baza*: $p_i \in F_P$, $i = 1,2,\ldots$
    - *inducţia*: dacă $U,V \in F_P$ atunci:

      $\neg U \in F_P$, $U \wedge V \in F_P$, $U \vee V \in F_P$, $U \rightarrow V \in F_P$, $U \leftrightarrow V \in F_P$
    - *închiderea*: toate formulele din F_P se obţin doar prin aplicarea regulilor precedente de un număr finit de ori.

Semantica logicii propoziţionale

- Propoziţiile logice sunt modele ale afirmaţiilor propoziţionale care sunt fie *adevărate*, fie *false*.

- **Scopul** definirii semanticii logicii propoziţionale este de a **atribui** un înţeles, **o valoare de adevăr**, formulelor propoziţionale.

- Domeniul semantic:

$$\{F(\text{fals}), T(\text{true, adevărat})\} \text{ a.î. } \neg F = T, \neg T = F$$

Semantica conectivelor

p	$\neg p$
T	F
F	T

$\uparrow$ - nand $p \uparrow q := \neg (p \wedge q)$

$\downarrow$ - nor $p \downarrow q := \neg (p \vee q)$

$\oplus$ - xor $p \oplus q := \neg (p \leftrightarrow q)$

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$	$p \uparrow q$	$p \downarrow q$	$p \oplus q$
T	T	T	T	T	T	F	F	F
T	F	F	T	F	F	T	F	T
F	T	F	T	T	F	T	F	T
F	F	F	F	T	T	T	T	F

Interpretarea (Def.)

- O ***interpretare*** a formulei $U(p_1,p_2,\ldots,p_n)\in F_P$ este o funcţie $i:\{p_1,p_2,\ldots,p_n\} \rightarrow\{T,F\}$ care asociază valori de adevăr variabilelor propoziţionale şi poate fi extinsă la o funcţie

  $i:F_P \rightarrow \{T,F\}$folosind relaţiile:

  $i(\neg p) = \neg i(p)$ $\quad$ $i(p\wedge q) = i(p) \wedge i(q)$

  $i(p\vee q) = i(p) \vee i(q)$ $\quad$ $i(p\rightarrow q) = i(p) \rightarrow i(q)$ $\quad$ $i(p\leftrightarrow q) = i(p)\leftrightarrow i(q)$
- Interpretările <u>evaluează</u> formulele propoziţionale conform semanticii conectivelor componente, atribuindu-le valori de adevăr.
- Tabela de adevăr a unei formule propoziţionale $U(p_1,p_2,\ldots,p_n)\in F_P$ corespunde evaluărilor formulei în <u>toate</u> cele 2^n interpretări.

Concepte semantice (Def.)

Fie formula propoziţională $U(p_1,p_2,\ldots,p_n) \in F_P$.

- O interpretare $i:\{p_1,p_2,\ldots,p_n\} \rightarrow \{T,F\}$ care evaluează formula U ca <u>adevărată</u>, $i(U)=T$, se numeşte ***model*** al formulei.

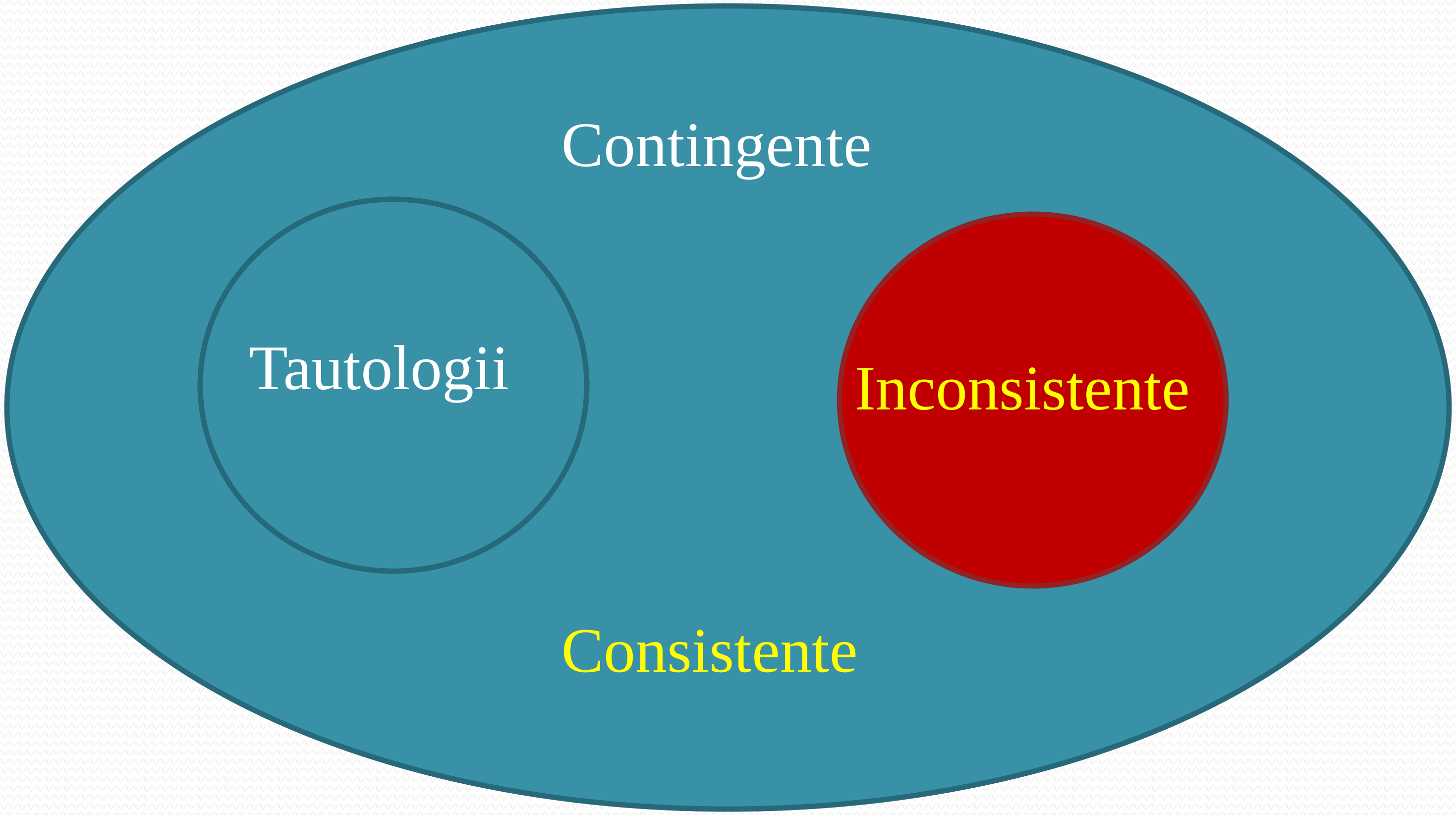
- O interpretare $i:\{p_1,p_2,\ldots,p_n\} \rightarrow \{T,F\}$ care evaluează formula U ca <u>falsă</u>, $i(U)=F$, se numeşte ***anti-model*** al formulei.

Concepte semantice (Def.) – cont.

Fie formula propoziţională $U(p_1,p_2,\ldots,p_n)\in F_P$.

- U se numeşte ***consistentă (realizabilă)*** dacă şi numai dacă are cel puţin un model, deci poate fi evaluată ca adevărată: $\exists i:\{p_1,p_2,\ldots,p_n\}\rightarrow\{T,F\}$ astfel încât $i(U)$=T.
- U se numeşte ***validă (tautologie)***, notaţie: $\models U$, dacă şi numai dacă U este evaluată ca adevărată în orice interpretare, adică: $\forall i:\{p_1,p_2,\ldots,p_n\}\rightarrow\{T,F\}$, $i(U)$=T. Toate interpretările formulei U sunt modele ale formulei.
- Formula U se numeşte ***inconsistentă (nerealizabilă)*** dacă şi numai dacă U nu are niciun model, adică U este interpretată totdeauna ca falsă: $\forall i:\{p_1,p_2,\ldots,p_n\}\rightarrow\{T,F\}$, $i(U)$=F.
- Formula U se numeşte ***contingentă*** dacă şi numai dacă este consistentă, dar nu este validă.

Tipuri de formule



Exemplu

- $U(p,q,r) = (\neg p \vee q) \wedge (r \vee p)$,
- $V(p,q,r) = (\neg p \wedge r) \vee (q \wedge r) \vee (q \wedge p)$
- $p \uparrow \neg p$
- $p \downarrow \neg p$

Tabela de adevăr - completarea

	p	q	r	$\neg p \vee q$	$r \vee p$	$U(p,q,r)$	$V(p,q,r)$	$p \uparrow \neg p$	$p \downarrow \neg p$
i_1	T	T	T	T	T	T	T	T	F
i_2	T	T	F	T	T	T	T	T	F
i_3	T	F	T	F	T	F	F	T	F
i_4	T	F	F	F	T	F	F	T	F
i_5	F	T	T	T	T	T	T	T	F
i_6	F	T	F	T	F	F	F	T	F
i_7	F	F	T	T	T	T	T	T	F
i_8	F	F	F	T	F	F	F	T	F

$U(p,q,r) = (\neg p \vee q) \wedge (r \vee p)$,

$V(p,q,r) = (\neg p \wedge r) \vee (q \wedge r) \vee (q \wedge p)$

Tabela de adevăr – interpretări

	p	q	r	$\neg p \vee q$	$r \vee p$	$U(p,q,r)$	$V(p,q,r)$	$p \uparrow \neg p$	$p \downarrow \neg p$
i_1	T	T	T	T	T	T	T	T	F
i_2	T	T	F	T	T	T	T	T	F
i_3	T	F	T	F	T	F	F	T	F
i_4	T	F	F	F	T	F	F	T	F
i_5	F	T	T	T	T	T	T	T	F
i_6	F	T	F	T	F	F	F	T	F
i_7	F	F	T	T	T	T	T	T	F
i_8	F	F	F	T	F	F	F	T	F

modele pt. U: i_1, i_2, i_5 şi i_7

$i_1 : \{p,q,r\} \rightarrow \{\mathrm{T},\mathrm{F}\}$, $i_1(p)=\mathrm{T}$, $i_1(q)=\mathrm{T}$, $i_1(r)=\mathrm{T}$ şi $i_1(U)=\mathrm{T}$

anti-modele pt. U: i_3, i_4, i_6 şi i_8

Tabela de adevăr – tip formule

	p	q	r	$\neg p \vee q$	$r \vee p$	$U(p,q,r)$	$V(p,q,r)$	$p \uparrow \neg p$	$p \downarrow \neg p$
i_1	T	T	T	T	T	T	T	T	F
i_2	T	T	F	T	T	T	T	T	F
i_3	T	F	T	F	T	F	F	T	F
i_4	T	F	F	F	T	F	F	T	F
i_5	F	T	T	T	T	T	T	T	F
i_6	F	T	F	T	F	F	F	T	F
i_7	F	F	T	T	T	T	T	T	F
i_8	F	F	F	T	F	F	F	T	F

$p \uparrow \neg p$ – tautologie

$p \downarrow \neg p$ – inconsistentă

U, V – contingente și consistente

Metasimboluri – relații semantice între formule

- Formula V este ***consecinţă logică*** a formulei U, notaţie: $U \models V$, dacă şi numai dacă $\forall i: F_P \rightarrow \{T,F\}$ astfel încât $i(U)=T$, are loc $i(V)=T$.

- Formulele $U(p_1,p_2,\ldots,p_n) \in F_P$ şi $V(p_1,p_2,\ldots,p_n) \in F_P$ sunt ***logic echivalente***, notaţie: $U \equiv V$, dacă şi numai dacă tabelele lor de adevăr sunt identice, adică: $\forall i: F_P \rightarrow \{T,F\}$, $i(U) = i(V)$.

Tabela de adevăr – tip formule

	p	q	r	$\neg p \vee q$	$r \vee p$	$U(p,q,r)$	$V(p,q,r)$	$p \uparrow \neg p$	$p \downarrow \neg p$
i_1	T	T	T	T	T	T	T	T	F
i_2	T	T	F	T	T	T	T	T	F
i_3	T	F	T	F	T	F	F	T	F
i_4	T	F	F	F	T	F	F	T	F
i_5	F	T	T	T	T	T	T	T	F
i_6	F	T	F	T	F	F	F	T	F
i_7	F	F	T	T	T	T	T	T	F
i_8	F	F	F	T	F	F	F	T	F

$U \equiv V$

$U \models \neg p \vee q$

Concepte semantice pentru mulțimi de formule

- O ***mulţime*** $\{U_1,U_2,\ldots,U_n\}$ de formule se numeşte ***consistentă (realizabilă)*** dacă şi numai dacă formula $U_1 \wedge U_2 \wedge \ldots \wedge U_n$ este consistentă, adică: $\exists i{:}F_P \rightarrow \{T,F\}$ astfel încât $i\ (U_1 \wedge U_2 \wedge \ldots \wedge U_n) = T$, ***i*** se numeşte ***model*** al mulţimii $\{U_1,U_2,\ldots,U_n\}$.
- O ***mulţime*** $\{U_1,U_2,\ldots,U_n\}$ de formule se numeşte ***inconsistentă (nerealizabilă, contradictorie)*** dacă şi numai dacă formula $U_1 \wedge U_2 \wedge \ldots \wedge U_n$ este inconsistentă, adică , $\forall i{:}F_P \rightarrow \{T,F\}$ astfel încât $i\ (U_1 \wedge U_2 \wedge \ldots \wedge U_n) = F$, ***i*** se numeşte ***anti-model*** al mulţimii $\{U_1,U_2,\ldots,U_n\}$.
- Formula V este ***consecinţă logică*** a mulţimii de formule $\{U_1,U_2,\ldots,U_n\}$ şi se notează $U_1,U_2,\ldots,U_n \models V$, dacă şi numai $\forall i{:}F_P \rightarrow \{T,F\}$ astfel încât $i\ (U_1 \wedge U_2 \wedge \ldots \wedge U_n) = T$, are loc $i\ (V) = T$. Formulele $U_1,U_2,\ldots,U_n$ se numesc *premize*, *ipoteze*, *fapte*, iar V se numeşte *concluzie*.

Teoremă

Fie $S=\{U_1,U_2,\ldots,U_n\}$ o mulţime de formule propoziţionale.

1. Dacă S este o *mulţime <u>consistentă</u>*, atunci $\forall j,\ 1\leq j\leq n,\ \ S \setminus \{U_j\}$ este o *mulţime <u>consistentă</u>*.
2. Dacă S este o *mulţime <u>consistentă</u>* şi V este o *formulă <u>validă</u>*, atunci *mulţimea* $S \cup \{V\}$ este *<u>consistentă</u>*.
3. Dacă S este o *mulţime <u>inconsistentă</u>*, atunci $\forall V \in \mathrm{F_p}$ *mulţimea* $S \cup \{V\}$ este *<u>inconsistentă</u>*.
4. Dacă S este o *mulţime <u>inconsistentă</u>* şi U_j este o *formulă <u>validă</u>*, unde $1\leq j\leq n$, atunci *mulţimea* $S \setminus \{U_j\}$ este *<u>inconsistentă</u>*.

Teoremă

Fie $U_1, U_2, \ldots, U_n$, U, V formule propoziţionale.

- $\models U$ ***dacă şi numai dacă*** $\neg U$ este *inconsistentă*
  (O formulă este *tautologie* ***dacă şi numai dacă*** *negaţia* sa este o formulă *inconsistentă).*
- $U \models V$ ***dacă şi numai dacă*** $\models U \to V$ ***dacă şi numai dacă*** mulţimea $\{U, \neg V\}$ este *inconsistentă.*
- $U \equiv V$ ***dacă şi numai dacă*** $\models U \leftrightarrow V$.
- $U_1, U_2, \ldots, U_n \models V$ ***dacă şi numai dacă*** $\models U_1 \wedge U_2 \wedge \ldots \wedge U_n \to V$ ***dacă şi numai dacă*** mulţimea $\{U_1, U_2, \ldots, U_n, \neg V\}$ este *inconsistentă.*

Echivalenţe logice în logica propoziţională

- Legile lui DeMorgan

  $\neg(U \wedge V) \equiv \neg U \vee \neg V$ și $\neg(U \vee V) \equiv \neg U \wedge \neg V$
- Legile de absorbţie

  $U \wedge (U \vee V) \equiv U$ și $U \vee (U \wedge V) \equiv U$
- Legile de comutativitate

  $U \wedge V \equiv V \wedge U$ și $U \vee V \equiv V \vee U$
- Legile de asociativitate

  $U \wedge (V \wedge Z) \equiv (U \wedge V) \wedge Z$ și $U \vee (V \vee Z) \equiv (U \vee V) \vee Z$
- Legile de distributivității

  $U \wedge (V \vee Z) \equiv (U \wedge V) \vee (U \wedge Z)$ și
  $U \vee (V \wedge Z) \equiv (U \vee V) \wedge (U \vee Z)$
- Legile de idempotență

  $U \wedge U \equiv U$ și $U \vee U \equiv U$

Alte echivalențe logice

- Legile de simplificare

$\neg\neg U \equiv U$ $\quad$ $U \to U \equiv \mathrm{T}$

$U \wedge \neg U \equiv \mathrm{F}$ $\quad$ $U \vee \neg U \equiv \mathrm{T}$

$\mathrm{T} \wedge U \equiv U$ $\quad$ $\mathrm{F} \vee U \equiv U$

$U \to \mathrm{T} \equiv \mathrm{T}$ $\quad$ $U \to \mathrm{F} \equiv \neg U$

$T \to U \equiv U$ $\quad$ $\mathrm{F} \to U \equiv \mathrm{T}$

$U \leftrightarrow \mathrm{T} \equiv U$ $\quad$ $U \leftrightarrow \mathrm{F} \equiv \neg U$

$U \oplus \mathrm{T} \equiv \neg U$ $\quad$ $U \oplus \mathrm{F} \equiv U$

$U \leftrightarrow U \equiv \mathrm{T}$ $\quad$ $U \oplus U \equiv \mathrm{F}$

- Definirea conectivelor

$U \to V \equiv \neg U \vee V$

$U \to V \equiv \neg(U \wedge \neg V)$

$U \to V \equiv U \leftrightarrow (U \wedge V)$

$U \to V \equiv V \leftrightarrow (U \vee V)$

$U \leftrightarrow V \equiv (U \to V) \wedge (V \to U)$

$U \oplus V \equiv \neg(U \to V) \vee \neg(V \to U)$

$U \leftrightarrow V \equiv (U \vee V) \to (U \wedge V)$

$U \vee V \equiv \neg(\neg U \wedge \neg V)$

$U \wedge V \equiv \neg(\neg U \vee \neg V)$

$U \vee V \equiv \neg U \to V$

$U \wedge V \equiv \neg(U \to \neg V)$

$\neg U \equiv U \uparrow U \equiv U \downarrow U$

$U \vee V \equiv (U \uparrow U) \uparrow (V \uparrow V) \equiv (U \downarrow V) \downarrow (U \downarrow V)$

$U \wedge V \equiv (U \downarrow U) \downarrow (V \downarrow V) \equiv (U \uparrow V) \uparrow (U \uparrow V)$

Principiul dualității

- Pentru orice echivalenţă logică $U \equiv V$ care conţine doar conectivele $\neg$, $\wedge$, $\vee$, $\uparrow$, $\downarrow$ există o altă echivalenţă logică, $U' \equiv V'$, unde U',V' sunt formule obţinute din U,V prin interschimbarea conectivelor logice duale: ($\wedge$, $\vee$), ($\uparrow$, $\downarrow$) şi a valorilor de adevăr: T, F.

- *Conective duale*: ($\wedge$, $\vee$) , ($\uparrow$, $\downarrow$), ($\leftrightarrow$, $\oplus$).
- *Valori de adevăr duale*: T şi F.
- *Concepte duale*: tautologie şi formulă inconsistentă.

Forme normale în logica propoziţiilor

1. Un ***literal*** este o variabilă propoziţională sau negaţia sa.
2. O ***clauză*** este disjuncţia unui număr finit de literali.
3. Un ***cub*** este conjuncţia unui număr finit de literali.
4. ***Clauza vidă***, simbolizată prin □, este clauza fără literali, fiind singura clauză inconsistentă.

5. O formulă este în ***formă normală disjunctivă*** (***FND***), dacă

   aceasta este scrisă ca o disjuncţie de cuburi $\vee_{i=1}^{p}(\wedge_{j=1}^{q_i} l_{ij})$

   unde l_{ij} sunt literali.

6. O formulă este în ***formă normală conjunctivă*** (***FNC***), dacă

   aceasta este scrisă ca o conjuncţie de clauze: $\wedge_{i=1}^{n}(\vee_{j=1}^{m_i} l_{ij})$

   unde l_{ij} sunt literali.

Exemple FN în logica propoziţiilor

- Un ***literal*** este o variabilă propoziţională sau negaţia sa. $p, \neg q$
- O ***clauză*** este disjuncţia unui număr finit de literali. $p \vee \neg q \vee r$
- Un ***cub*** este conjuncţia unui număr finit de literali. $p \wedge \neg q \wedge \neg r$
- O formulă este în ***formă normală disjunctivă*** **(*FND*)**, dacă aceasta este scrisă ca o disjuncţie de cuburi.

$$(\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (p \wedge q \wedge \neg r)$$

- O formulă este în ***formă normală conjunctivă*** **(*FNC*)**, dacă aceasta este scrisă ca o conjuncţie de clauze.

$$(\neg p \vee q \vee r) \wedge (\neg p \vee \neg r) \wedge (p \vee q \vee \neg r) \wedge (p \vee q)$$

Sunt în FNC și/sau FND?

$(\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee \neg r)$ FNC, 3 clauze

$(\neg p \wedge q \wedge r) \vee (\neg p \wedge \neg r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge q)$ FND, 4 cuburi

p FNC, 1 clauză; FND, 1 cub

$\neg q$ FNC, 1 clauză; FND, 1 cub

$\neg p \vee q \vee \neg r$ FND, 3 cuburi; FNC, 1 clauză

$p \wedge \neg q \wedge \neg r$ FNC, 3 clauze; FND, 1 cub

Proprietate

Fie mulţimea de literali $\{l_1, l_2, \ldots, l_n\}$. Următoarele afirmaţii sunt echivalente:

- clauza $\vee_{i=1}^{n} l_i$ este validă;
- cubul $\wedge_{i=1}^{n} l_i$ este inconsistent;
- în mulţimea $\{l_1, l_2, \ldots, l_n\}$ există cel puţin o pereche de literali opuşi, adică: $\exists i, j \in \{1, \ldots, n\}$ astfel încât $l_i = \neg\, l_j$.

Proprietate - exemple

Fie mulţimea de literali $\{l_1, l_2, \ldots, l_n\}$. Următoarele afirmaţii sunt echivalente:

- clauza $\vee_{i=1}^{n} l_i$ este validă;
- cubul $\wedge_{i=1}^{n} l_i$ este inconsistent;
- în mulţimea $\{l_1, l_2, \ldots, l_n\}$ există cel puţin o pereche de literali opuşi, adică: $\exists i, j \in \{1, \ldots, n\}$ astfel încât $l_i = \neg\, l_j$.

$$\neg p \vee q \vee p \equiv \mathrm{T}$$

$$p \wedge r \wedge \neg r \equiv \mathrm{F}$$

Teoremă

- *Orice formulă* admite *o formă normală conjunctivă* şi *o formă normală disjunctivă* logic echivalente cu ea.

Algoritmul de normalizare

Pas1:

Înlocuirea formulelor de tip $\boldsymbol{U \to V}$ cu forma echivalentă: $\neg U \vee V$

Înlocuirea formulelor de tip $\boldsymbol{U \leftrightarrow V}$ cu forma echivalentă:

$$(\neg U \vee V) \wedge (\neg V \vee U).$$

Pas2:

Aplicarea legilor lui **DeMorgan** (se recomandă aplicarea dinspre exterior spre interior) ==> negaţia va preceda doar variabilele propoziţionale.

Eliminarea negaţiilor multiple folosind echivalenţa logică: $\neg\neg U \equiv U$.

Pas3: Aplicarea legilor **distributivităţii**.

Pentru FND FNC

$U \wedge (V \vee Z) \equiv (U \wedge \mathrm{V}) \vee (U \wedge Z)$ respectiv $U \vee (V \wedge Z) \equiv (U \vee \mathrm{V}) \wedge (U \vee Z)$

Pas4: Simplificarea formei obţinute folosind alte echivalenţe logice: legile de simplificare, legile absorbţiei, legile de idempotenţă.

Exemplu de aducere la FNC & FND

$U = p \to r \lor \neg(\neg q \lor r)$

$U = p \to (r \lor \neg(\neg q \lor r))$

Pas1: Înlocuirea formulelor de tip $U \leftrightarrow V$, $\boldsymbol{U \to V}$ cu forma echivalentă: …, $\neg U \lor V$

$U \equiv \neg p \lor (r \lor \neg(\neg q \lor r))$

Pas2: Aplicarea legilor lui **DeMorgan**

$U \equiv \neg p \lor (r \lor (\neg\neg q \land \neg r))$

$U \equiv \neg p \lor (r \lor (q \land \neg r))$

$U \equiv \neg p \lor r \lor (q \land \neg r)$ – *FND*

Pas3: Aplicarea legilor **distributivităţii**

$U \equiv (\neg p \lor r \lor q) \land (\neg p \lor \underline{r} \lor \underline{\neg r})$ – *FNC*

Pas4: Simplificarea formei obţinute folosind alte echivalenţe logice

$U \equiv \neg p \lor r \lor q$ – *FND & FNC*

Teoremă

- O *formulă* în *forma normală conjunctivă* (*FNC*) este *tautologie* dacă şi numai dacă *toate clauzele* sale sunt *valide.*
- O *formulă* în *forma normală disjunctivă* (*FND*) este *inconsistentă* dacă şi numai dacă *toate cuburile* sale sunt *inconsistente.*

Observații

- Prima parte a teoremei furnizează o ***metodă directă*** de rezolvare a problemei decizionale (verificarea dacă o formulă este tautologie) în logica propoziţiilor.
- FND a unei formule propoziţionale furnizează toate <u>modelele</u> formulei iniţiale, prin găsirea interpretărilor care evaluează cuburile componente ca adevărate.
- FNC a unei formule propoziţionale furnizează toate <u>anti-modelele</u> formulei iniţiale prin găsirea interpretărilor care evaluează clauzele componente ca false.
- Cele două formulări din teorema precedentă sunt afirmaţii duale.

Logică computațională
Curs 4

Lector dr. Pop Andreea-Diana

Sisteme axiomatice

- Axiomele geometriei
- Axiomele aritmeticii

Sistemul axiomatic al calculului propoziţional

- propus de Hilbert; deductiv, formal
- $P=(\Sigma_P , F_P , A_P , R_P)$
  - $\Sigma_P = Var\_propoz \cup Conective \cup \{ (,) \}$
    - $Var\_propoz = \{ p, q, r, p_1, p_2, \dots\}$
    - $Conective = \{ \neg , \wedge , \vee , \rightarrow , \leftrightarrow \}$
  - F_P = mulţimea formulelor propoziţionale corect construite
    - - *baza*: $p_i \in F_P$, $i=1,2,\dots$
    - - *inducţia*: dacă $U,V \in F_P$ atunci:

      $\neg U \in F_P$, $U \wedge V \in F_P$, $U \vee V \in F_P$, $U \rightarrow V \in F_P$, $U \leftrightarrow V \in F_P$
    - - *închiderea*: toate formulele din F_P se obţin doar prin aplicarea regulilor precedente de un număr finit de ori.

Axiome și reguli de inferență

- $A_P = \{A_1 , A_2 , A_3\}$ scheme axiomatice
  - A_1: $U \to (V \to U)$
  - A_2: $(U \to (V \to Z)) \to ((U \to V) \to (U \to Z))$
  - A_3: $(U \to V) \to (\neg V \to \neg U)$
- $R_P = \{m_P\}$ – o singură regulă de inferență „ *modus ponens*”
  - $U , U \to V \vdash V$

„din faptele U şi $U \to V$ se deduce (inferă) V”

Definiția deducției

- Fie formulele $U_1, U_2, \dots, U_n$ numite ipoteze şi V formulă propoziţională. Spunem că **V este deductibilă din $U_1, U_2, \dots, U_n$** şi notăm $U_1, U_2, \dots, U_{n-1}, U_n \vdash V$, dacă există o secvenţă de formule $(f_1, f_2, \dots, f_m)$ astfel încât $f_m = V$ şi $\forall i \in \{1,\dots,m\}$ avem:
  - $f_i \in A_{\mathrm{p}}$;
  - $f_i \in \{U_1, U_2, \dots, U_n\}$;
  - $f_j, f_k \vdash_{m_{\mathrm{p}}} f_i$, $j < i$ și $k < i$
- Secvenţa $(f_1, f_2, \dots, f_m)$ se numeşte ***deducţia* lui V din $U_1, U_2, \dots, U_n$**.

Noțiunea de teoremă

- **Definiţia 1.8.** O formulă $U \in F_P$, astfel încât $\emptyset \vdash U$ (sau $\vdash U$) se numeşte ***teoremă***.

- **Observaţie**: Teoremele sunt formule care sunt deductibile doar din axiome şi folosind regula modus ponens.

Teorema de deducție și inversa sa

- **Teorema de deducţie**

  Dacă $U_1, U_2, \ldots, U_{n-1}, U_n \vdash V$, atunci $U_1, U_2, \ldots, U_{n-1} \vdash U_n \rightarrow V$.

- **Inversa teoremei de deducţie**

  Dacă $U_1, U_2, \ldots, U_{n-1} \vdash U_n \rightarrow V$ atunci $U_1, U_2, \ldots, U_{n-1}, U_n \vdash V$.

Generalizarea

$U_1, U_2, \ldots, U_{n-1}, U_n \vdash V$ dacă și numai dacă

$U_1, U_2, \ldots, U_{n-1} \vdash U_n \to V$ dacă și numai dacă

$U_1, U_2, \ldots, U_{n-2} \vdash U_{n-1} \to (U_n \to V)$ dacă și numai dacă

$\ldots$

$U_1 \vdash U_2 \to (\ldots U_{n-1} \to (U_n \to V) \ldots)$ dacă și numai dacă

$\vdash U_1 \to (U_2 \to (\ldots U_{n-1} \to (U_n \to V \underbrace{)\ldots)}_{n-1}$

Consecințele teoremei de deducție

- $\vdash U \to ((U \to V) \to V)$
- $\vdash (U \to V) \to ((V \to Z) \to (U \to Z))$ legea silogismului
- $\vdash (U \to (V \to Z)) \to (V \to (U \to Z))$ legea permutării premizelor
- $\vdash (U \to (V \to Z)) \to (U \wedge V \to Z)$ legea reuniunii premizelor
- $\vdash (U \wedge V \to Z) \to (U \to (V \to Z))$ legea separării premizelor

Proprietăţile logicii propoziţiilor

- **Problemele decizionale** în logica propoziţiilor:
  - „Este o formulă propoziţională o teoremă sau nu?”
  - „Este o formulă deductibilă dintr-o mulţime de formule?”

- **Teorema de corectitudine**

  Dacă $\vdash U$ **atunci** $\models U$. (Validitatea sintactică implică validitatea semantică)
- **Teorema de completitudine**

  Dacă $\models U$ **atunci** $\vdash U$. (Validitatea semantică implică validitatea sintactică)
- **Teorema de corectitudine și completitudine**

  $\vdash U$ **dacă și numai dacă** $\models U$.

Consecințele teoremei de corectitudine și completitudine

1) ***Logica propoziţiilor este necontradictorie:***

   nu pot avea loc simultan $\vdash U$ şi $\vdash \neg U$.

2) ***Logica propoziţiilor este coerentă:***

   nu orice formulă propoziţională este teoremă.

3) ***Logica propoziţiilor este decidabilă:***

   se poate decide dacă o formulă propoziţională este sau nu teoremă.

Logică computațională
Curs 5

Lector dr. Pop Andreea-Diana

Metoda tabelelor semantice

- introdusă de Smullyan
- se bazează pe considerații semantice
- încearcă să construiască modelele unei formule date

(FND)

- $\models U$ prin respingere, $\neg U$ nu are modele

- ideea:
  - descompunerea formulei inițiale în subformule
  - până la nivel de literali

Clase de formule

- clasa α - formule de tip conjunctiv

$$A \wedge B$$

$$\neg (A \vee B)$$

$$\neg (A \rightarrow B)$$

- clasa β - formule de tip disjunctiv

$$A \vee B$$

$$\neg (A \wedge B)$$

$$A \rightarrow B$$

Reguli de descompunere a formulelor

- regula α

$$\begin{array}{c} A \wedge B \\ | \\ A \\ | \\ B \end{array} \qquad \begin{array}{c} \neg (A \vee B) \\ | \\ \neg A \\ | \\ \neg B \end{array} \qquad \begin{array}{c} \neg (A \rightarrow B) \\ | \\ A \\ | \\ \neg B \end{array}$$

- regula β

$$\begin{array}{ccc} & A \vee B & \\ / & & \backslash \\ A & & B \end{array}$$

$$\begin{array}{ccc} & \neg (A \wedge B) & \\ / & & \backslash \\ \neg A & & \neg B \end{array}$$

$$\begin{array}{ccc} & A \rightarrow B & \\ / & & \backslash \\ \neg A & & B \end{array}$$

Arborele binar de descompunere a unei formule

Având o formulă *U*, ei i se poate asocia o tabelă semantică, care este de fapt un arbore binar ce conţine în nodurile sale formule şi se construieşte astfel:

- rădăcina arborelui este etichetată cu formula *U*;
- fiecare ramură a arborelui care conţine o formulă va fi extinsă cu subarborele corespunzător regulii de descompunere care se aplică formulei;
- extinderea unei ramuri se *încheie* în două situaţii:
  a) dacă pe ramură apare o formulă şi negaţia sa;
  b) dacă au fost descompuse toate formulele de pe acea ramură

Tipuri de ramuri

- O *ramură* a tabelei se numeşte *închisă* (simbolizată prin ⊗) dacă ea conţine o formulă şi negaţia ei, în caz contrar *ramura* se numeşte *deschisă* (simbolizată prin⊙).
- O *ramură* a tabelei se numeşte *completă* dacă ea este fie *închisă*, fie *toate formulele* de pe acea ramură au fost *descompuse*.

Tipuri de tabele semantice

- O *tabelă* se numeşte *închisă* dacă toate ramurile sale sunt închise. Dacă o tabelă are cel puţin o ramură deschisă, atunci ea se numeşte *deschisă*.
- O *tabelă* se numeşte *completă* dacă toate ramurile ei sunt complete.

Exemplul (2)

$U = (p \vee q) \wedge \neg(q \to r) \wedge (p \to q \wedge r)$ (1)

| – regula α pentru (1)

$p \vee q$ (2)

$\neg(q \to r)$ (3)

$p \to q \wedge r$ (4)

– regula β pentru (2)

Left branch:

$\boxed{p}$

regula α pentru (3) –

q

$\boxed{\neg r}$

regula β pentru (4) –

⊗ $\boxed{\neg p}$ | $q \wedge r$ (5)

regula α pentru (5) –

q

$\boxed{r}$

⊗

Right branch:

q

– regula α pentru (3)

q

$\boxed{\neg r}$

– regula β pentru (4)

$\neg p$ | $q \wedge r$ (6)

⊙

– regula α pentru (6)

q

$\boxed{r}$

⊗

Observaţii:

- Procesul de construire a unei tabele semantice este unul *nedeterminist* deoarece regulile de descompunere se pot aplica în orice ordine şi la un moment dat se pot alege mai multe ramuri pentru extindere. Astfel unei formule i se pot asocia mai multe tabele semantice, dar acestea sunt echivalente.
- Pentru a obţine tabele semantice *cât mai simple* (mai puţin ramificate) se recomandă:
  - utilizarea regulilor de tip α înaintea regulilor de tip β care realizează o ramificare;

Observaţii (2):

- formulele de pe aceeaşi ramură a unei tabele semantice sunt *legate* între ele prin conectiva logică $\wedge$, iar *ramificarea* corespunde conectivei logice $\vee$.
- tabela semantică asociată unei formule propoziţionale este o reprezentare grafică a *formei* sale *normale disjunctive*. Fiecare ramură reprezintă un *cub* (conjuncţia tuturor literalilor de pe acea ramură), iar arborele este *disjuncţia* tuturor *ramurilor* sale.
- Unei formule *consistente* i se asociază o *tabelă completă deschisă*, iar fiecare *ramură deschisă* a tabelei furnizează cel puţin un *model* pentru formula respectivă.
- O *tabelă semantică închisă* asociată unei formule indică faptul că formula este *inconsistentă*, adică nu există nicio interpretare în care formula să fie adevărată

Teorema de corectitudine şi completitudine a metodei tabelelor semantice

- O formulă U este teoremă (tautologie) dacă şi numai dacă există o tabelă semantică închisă pentru formula $\neg U$.

Teoremă

- $U_1, U_2, ..., U_n \vdash Y$ (echivalent cu $U_1, U_2, ..., U_n \models Y$) dacă şi numai dacă există o tabelă semantică închisă pentru formula $U_1 \wedge U_2 \wedge ... \wedge U_n \wedge \neg Y$.

Logică computațională
Curs 6

Lector dr. Pop Andreea-Diana

Metoda rezoluţiei (Robinson, 1965)

- metodă de demonstrare automată *sintactică*, prin *respingere*
- este o metodă corectă şi completă de demonstrare automată
- verificarea *consistenţei/inconsistenţei* unei mulţimi de clauze (scop)

Sistem formal (axiomatic) asociat Rezoluției propoziționale

- Res = $(\Sigma_{\mathrm{Res}}, F_{\mathrm{Res}}, A_{\mathrm{Res}}, R_{\mathrm{Res}})$
  - $\Sigma_{\mathrm{Res}} = \Sigma_{\mathrm{P}} \setminus \{\wedge, \rightarrow, \leftrightarrow\}$ – **alfabetul**
  - $F_{\mathrm{Res}} \cup \{\square\}$ – **mulţimea formulelor bine-formate**
    - F_{Res} mulţimea tuturor clauzelor ce se pot forma folosind alfabetul Σ_{Res}
    - $\square$ - **clauza vidă** care nu conţine nici un literal, simbolizează inconsistenţa
  - $A_{\mathrm{Res}} = \varnothing$ – **mulțimea axiomelor**
  - $R_{\mathrm{Res}} = \{res\}$ **mulţimea regulilor de inferenţă** care conţine doar
    - **regula rezoluţiei**: $A \vee l, B \vee \neg l \vdash_{res} A \vee B$, unde l este un literal, iar $A, B \in F_{\mathrm{Res}}$

Terminologie

- clauzele $C_1 = A \vee l$, $C_2 = B \vee \neg l$ **rezolvă** deoarece conţin doi literali opuşi (complementari)
- **Notație:** $C_3 = Res_l\,(C_1, C_2)$
- C_3 **rezolventul** clauzelor C_1 și C_2
- clauzele C_1, C_2 **clauze părinte**
- caz particular: $C_1 = l$, $C_2 = \neg l$, $Res_l\,(C_1, C_2) = \square$ - inconsistentă

Observație:

- Rezoluţia ca şi regulă de inferenţă este o generalizare a regulilor *modus ponens*, *modus tollens* şi a *silogismului*.

- *mp:* $U, U \to V \vdash V \quad \Rightarrow \quad U, \neg U \vee V \vdash V$
- *mt:* $U \to V \vdash \neg V \to \neg U \overset{\text{ITD}}{\Rightarrow} U \to V, \neg V \vdash \neg U \Rightarrow$

  $\neg U \vee V, \neg V \vdash \neg U$
- *sil.:* $U \to V, V \to Z \vdash U \to Z \Rightarrow$

  $\neg U \vee V, \neg V \vee Z \vdash \neg U \vee Z$

Algoritmul rezoluției propoziționale:

Date de intrare: S – o mulţime de clauze
Date de ieșire: S consistentă sau inconsistentă

$S_0 = S$
$i = 0$
Repetă
@ se aleg două clauze $C_1, C_2 \in S$ care rezolvă
$C_3 = \text{Res}(C_1, C_2)$
$S_{i+1} = S_i \cup \{C_3\}$
Dacă $C_3 = \square$
Atunci Scrie ”S este inconsistentă”; **STOP**
Altfel $i = i + 1$
Sfârșit_dacă
Până când $S_i = S_{i-1}$ *//nu se mai pot deriva clauze noi*
Scrie ”S este consistentă”
Sfârșit algoritm

Notație:

- $S \vdash_{\mathrm{Res}} \square$ ”din mulțimea S de clauze s-a derivat clauza vidă prin aplicarea algoritmului rezoluției propoziținale”

Teorema de corectitudine și completitudine

- Teorema de corectitudine

  Dacă $S \vdash_{\text{Res}} \square$ atunci S este inconsistentă.

- Teorema de completitudine

  Dacă S este inconsistentă atunci $S \vdash_{\text{Res}} \square$.

- Teorema de corectitudine și completitudine

  Mulțimea S este inconsistentă dacă și numai dacă $S \vdash_{\text{Res}} \square$.

Teoreme

- U este tautologie dacă și numai dacă $FNC\,(\neg\, U\,) \vdash_{\text{Res}} \square$
- $U_1, U_2, \ldots, U_n \models V$ dacă și numai dacă

  $U_1, U_2, \ldots, U_n \vdash V$ dacă și numai dacă

  $FNC\,(U_1 \wedge U_2 \wedge \ldots \wedge U_n \wedge \neg V\,) \vdash_{\text{Res}} \square$

  $S_i \stackrel{\text{not.}}{=} FNC\,(U_i\,),\ i = \overline{1,n}$

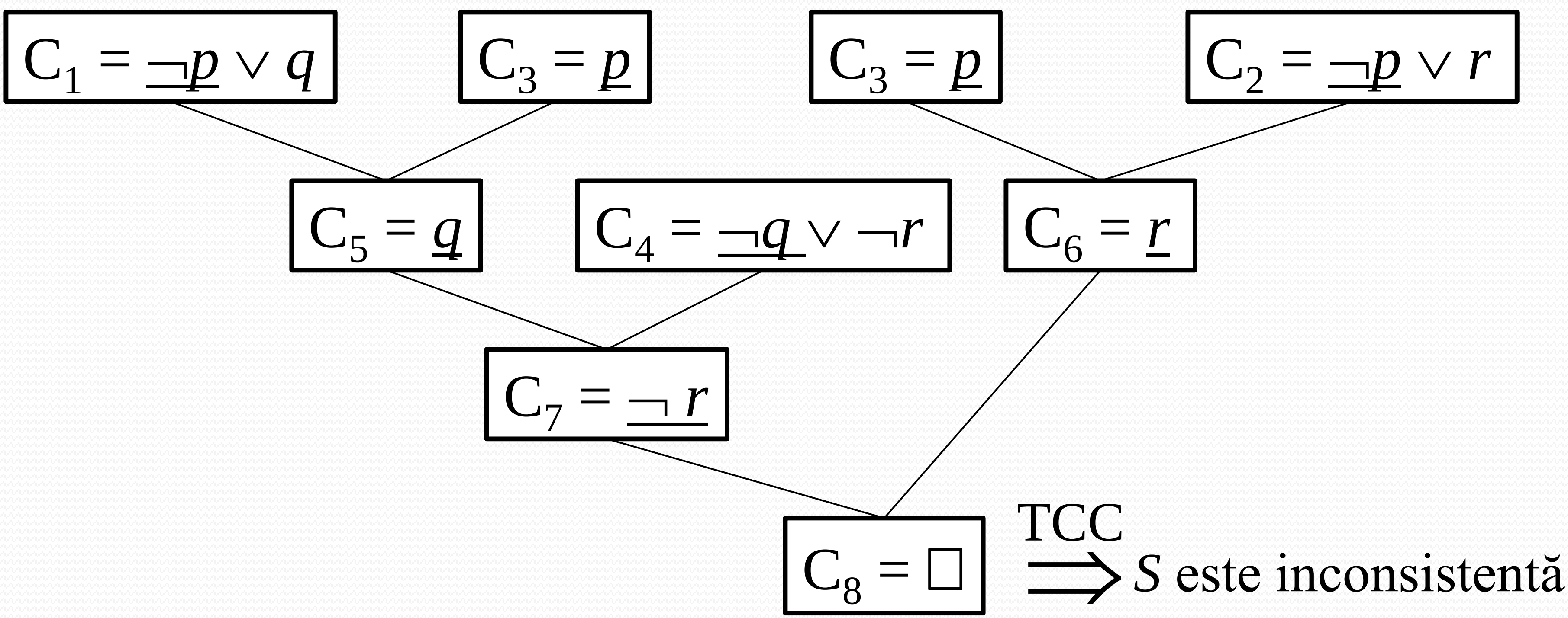
  $S_{n+1} \stackrel{\text{not.}}{=} FNC\,(\neg V\,)$

  $S_1 \cup S_2 \cup \ldots \cup S_n \cup S_{n+1} \vdash_{\text{Res}} \square$

Exemplu

$S = \{\neg p \vee q, \neg p \vee r, p, \neg q \vee \neg r\}$

$C_1 \overset{\text{not.}}{=\!=} \neg p \vee q, \quad C_2 \overset{\text{not.}}{=\!=} \neg p \vee r, \quad C_3 \overset{\text{not.}}{=\!=} p, \quad C_4 \overset{\text{not.}}{=\!=} \neg q \vee \neg r$



Problemă propusă

Ion, Vasile și Gheorghe sunt studenți la Informatică. Ei au făcut următorul pact:

- Dacă Ion își face laboratorul atunci Ion sau Vasile au laboratorul făcut.
- Dacă Ion își face laboratorul, atunci și Vasile își face laboratorul.
- Dacă Vasile nu-și face laboratorul, atunci Gheorghe îl va face.
- Dacă Gheorghe nu își face laboratorul, atunci nu își face nici Ion laboratorul sau Vasile îl face.
- Știm că Gheorghe nu și-a făcut laboratorul.

1. Să se verifice dacă pactul nu este contradictoriu
2. Să se verifice dacă din cele de mai sus se poate deduce că Vasile a făcut laboratorul.

Notații

Ion și-a făcut lab. $\overset{\text{not.}}{=}$ p,

Vasile și-a făcut lab. $\overset{\text{not.}}{=}$ q,

Gheorghe și-a făcut lab. $\overset{\text{not.}}{=}$ r.

- Dacă Ion își face laboratorul atunci Ion sau Vasile au laboratorul făcut.
- Dacă Ion își face laboratorul, atunci și Vasile își face laboratorul.
- Dacă Vasile nu-și face laboratorul, atunci Gheorghe îl va face.
- Dacă Gheorghe nu își face laboratorul, atunci nu își face nici Ion laboratorul sau Vasile îl face.
- Știm că Gheorghe nu și-a făcut laboratorul.

? Vasile a făcut laboratorul.

Transformare limbaj natural în limbaj logic

Ion și-a făcut lab. $\overset{\text{not.}}{=} p$,
Vasile și-a făcut lab. $\overset{\text{not.}}{=} q$,
Gheorghe și-a făcut lab. $\overset{\text{not.}}{=} r$.

- Dacă Ion își face laboratorul atunci Ion sau Vasile au laboratorul făcut. $p \to p \vee q$
- Dacă Ion își face laboratorul, atunci și Vasile își face laboratorul $p \to q$
- Dacă Vasile nu-și face laboratorul, atunci Gheorghe îl va face. $\neg q \to r$
- Dacă Gheorghe nu își face laboratorul, atunci nu își face nici Ion laboratorul sau Vasile îl face. $\neg r \to \neg p \vee q$
- Știm că Gheorghe nu și-a făcut laboratorul. $\neg r$

? Vasile a făcut laboratorul. q

În limbaj logic

- $p \to p \vee q$
- $p \to q$
- $\neg q \to r$
- $\neg r \to \neg p \vee q$
- $\neg r$

$? \, q$

Obținerea mulțimilor de clauze

- $p \to p \vee q \equiv \neg p \vee p \vee q \overset{\text{not.}}{=} C_1$
- $p \to q \equiv \neg p \vee q \overset{\text{not.}}{=} C_2$
- $\neg q \to r \equiv q \vee r \overset{\text{not.}}{=} C_3$
- $\neg r \to \neg p \vee q \equiv r \vee \neg p \vee q \overset{\text{not.}}{=} C_4$
- $\neg r \overset{\text{not.}}{=} C_5$

$$?\, q \quad \overset{\text{se neagă concluzia}}{\Longrightarrow} \quad \neg q \overset{\text{not.}}{=} C_6$$

$S_1 = \{C_1, C_2, C_3, C_4, C_5\}$

$S_2 = \{C_1, C_2, C_3, C_4, C_5, C_6\}$

Aplicarea metodei rezoluției

$S_1 = \{\quad C_1,\quad C_2,\quad C_3,\quad C_4,\quad C_5\} =$

$= \{\neg p \vee p \vee q\,,\ \neg p \vee q\,,\ q \vee r\,,\ r \vee \neg p \vee q\,,\ \neg r\}$

$S_2 = \{\quad C_1,\quad C_2,\quad C_3,\quad C_4,\quad C_5,\quad C_6\} =$

$= \{\neg p \vee p \vee q\,,\ \neg p \vee q\,,\ q \vee r\,,\ r \vee \neg p \vee q\,,\ \neg r\,,\ \neg q\}$

Aplicarea metodei rezoluției pt. S_1:

$S_1 = \{ \quad C_1, \quad C_2, \quad C_3, \quad C_4, \quad C_5 \} =$

$= \{ \neg p \vee p \vee q \,, \neg p \vee q \,, q \vee r \,, r \vee \neg p \vee q \,, \neg r \}$

$C_6 = \mathrm{Res}_r\,(C_3, C_5) = q$

$\cancel{C_2 = \mathrm{Res}_p\,(C_1, C_2) = \neg p \vee q}$ nu s-a obținut o clauză nouă

$\cancel{C_1 = \mathrm{Res}_p\,(C_1, C_1) = \neg p \vee p \vee q}$ nu s-a obținut o clauză nouă

$\cancel{C_2 = \mathrm{Res}_r\,(C_4, C_5) = \neg p \vee q}$ nu s-a obținut o clauză nouă

Abandonăm fără a ajunge la o concluzie, am constatat că avem nevoie de o strategie.

Deci nu știm încă dacă ipotezele sunt contradictorii.

Aplicarea metodei rezoluției pt. S_2:

S_2= { C_1, C_2, C_3, C_4, C_5, C_6}=

$$=\{\neg p \vee p \vee q\ , \neg p \vee q\ , q \vee r\ , r \vee \neg p \vee q\ , \neg r\ , \neg q\}$$

C_7= Res_r (C_3, C_5) = q TCC

C_8= Res_r (C_7, C_6) = $\square \Rightarrow S_2$ e inconsistentă adică, Vasile își face laboratorul

Automatizarea procesului rezolutiv

- prin intermediul unor *strategii*
  - asigură exploatarea tuturor modurilor posibile de derivare a clauzei vide
  - evitarea deducerii unor clauze redundante sau irelevante pentru obținerea $\square$

Strategia eliminării

- inspirată din procedura Davis-Putman
- O mulțime *S* de clauze poate fi simplificată, păstrând consistența/inconsistența ei prin aplicarea următoarelor transformări:
  - **Eliminarea clauzelor tautologice** (nu pot contribui la derivarea clauzei vide): $\cancel{\underline{\neg p} \vee q \vee \underline{p} \vee \neg r}$
  - **Eliminarea clauzelor subsumate de alte clauze din *S***: clauza C_1 este subsumată de C_2 dacă există o clauză C_3 astfel încât $C_1 = C_2 \vee C_3$:

    $\cancel{\underline{\neg p \vee q} \vee r}$ este subsumată de $\neg p \vee q$
  - **Eliminarea clauzelor care conțin literali puri în *S***: Un literal este pur dacă negația sa nu apare în nici o clauză din *S*:

    $\{\neg p \vee q, \cancel{\neg p \vee \underline{r}}, p, \cancel{\neg q \vee \underline{r}}\}$

- Dacă C=*l* este o **clauză unitate** din *S*, **se șterg toate clauzele *care-l conțin pe l* și *¬l din clauzele rămase*.**

$\{\cancel{\underline{\neg p}} \vee q, \cancel{\underline{\neg p}} \vee r, \cancel{p}, \neg q \vee \neg r, \cancel{p \vee \neg r}\} \Rightarrow \{q, r, \neg q \vee \neg r\}$

 ∅ sau {□} 

Strategia eliminării aplicată la S_1:

$S_1 = \{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r\}$

- **Eliminarea clauzelor tautologice** (nu pot contribui la derivarea clauzei vide):

$S_1 = \{\neg p \vee q, q \vee r, \boldsymbol{r \vee \neg p \vee q}, \neg r\}$

- **Eliminarea clauzelor subsumate de alte clauze din *S***: clauza C_1 este subsumată de C_2 dacă există o clauză C_3 astfel încât $C_1 = C_2 \vee C_3$:

$S_1 = \{\neg p \vee q, q \vee r, \neg r\}$

- **Eliminarea clauzelor care conțin literali puri în *S***: Un literal este pur dacă negația sa nu apare în nici o clauză din *S*:

$S_1 = \{q \vee r, \neg r\}$

$S_1 = \{\neg r\}$

$S_1 = \{\} = \varnothing$ - consistentă, deci pactul studenților este necontradictoriu

Strategia eliminării aplicată la S_2:

$S_2 = \{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r, \neg q\}$

- **Eliminarea clauzelor tautologice** (nu pot contribui la derivarea clauzei vide):

$S_2 = \{\neg p \vee q, q \vee r, \mathbf{r \vee \neg p \vee q}, \neg r, \neg q\}$

- **Eliminarea clauzelor subsumate de alte clauze din *S***: clauza C_1 este subsumată de C_2 dacă există o clauză C_3 astfel încât $C_1 = C_2 \vee C_3$:

$S_2 = \{\neg p \vee q, q \vee r, \neg r, \neg q\}$

- **Eliminarea clauzelor care conțin literali puri în *S***: Un literal este pur dacă negația sa nu apare în nici o clauză din *S*:

$S_2 = \{q \vee r, \neg r, \neg q\}$

- Dacă C=*l* este o **clauză unitate** din *S*, **se șterg toate clauzele *care-l conțin pe l* și *¬l din clauzele rămase***.

$S_2 = \{q, \neg q \vee \square\}$

$S_2 = \{\square\}$ – inconsistentă, deci Vasile și-a făcut laboratorul

Strategia saturării pe nivele (algoritmul)

Date de intrare: S – o mulţime de clauze

Date de ieșire: S consistentă sau inconsistentă

//Se generează mulțimile de clauze S^0 , S^1, … S^k ce reprezintă nivelele

$S^0 = S$

$k = 0$

Repetă

$k = k + 1$

$S^k = \{\mathrm{Res}\ (C_1, C_2) \mid C_1 \in S^0 \cup S^1 \cup \ldots \cup S^{k-1} , C_2 \in S^{k-1} \}$

$S^k = S^k \setminus (S^0 \cup S^1 \cup \ldots \cup S^{k-1})$

Până când $\square \in S^k$ **sau** $S^k = \varnothing$

Dacă $\square \in S^k$

Atunci Scrie ”S este inconsistentă”;

Altfel Scrie ”S este consistentă”

Sfârșit_dacă

Sfârșit algoritm

Strategia saturării pe nivele aplicată la S_1 (1):

$S_1^0 = \{ \quad C_1, \quad C_2, \quad C_3, \quad C_4, \quad C_5 \} =$

$= \{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r\}$

$C_2 = Res_p (C_1, C_2) = \neg p \vee q$

$Res_? (C_1, C_3)$ NU

$C_4 = Res_p (C_1, C_4) = \neg p \vee q \vee r$

$Res_? (C_1, C_5)$ NU

$Res_? (C_2, C_3)$ NU

$Res_? (C_2, C_4)$ NU

$Res_? (C_2, C_5)$ NU

$Res_? (C_3, C_4)$ NU

$C_6 = Res_r (C_3, C_5) = q$

$C_2 = Res_r (C_4, C_5) = \neg p \vee q$

$S_1^1 = \{C_6\} = \{q\}$

Strategia saturării pe nivele aplicată la S_1 (2):

$S_1^0 = \{ C_1, C_2, C_3, C_4, C_5\}=$

$=\{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r\}$

$S_1^1= \{C_6\}=\{q\}$

$Res_? (C_6, C_1)$ NU

$Res_? (C_6, C_2)$ NU

$Res_? (C_6, C_3)$ NU

$Res_? (C_6, C_4)$ NU

$Res_? (C_6, C_5)$ NU

$S_1^2= \{\}=\emptyset \Rightarrow S_1$ este consistent

Strategia saturării pe nivele aplicată la S_2 (1):

$S_2 = \{ \quad C_1, \quad C_2, \quad C_3, \quad C_4, \quad C_5, \quad C_6 \} =$

$= \{ \neg p \vee p \vee q \,,\, \neg p \vee q \,,\, q \vee r \,,\, r \vee \neg p \vee q \,,\, \neg r \,,\, \neg q \}$

$C_2 = \mathrm{Res}_p\, (C_1, C_2) = \neg p \vee q$

$\mathrm{Res}_?\, (C_1, C_3)$ NU

$C_4 = \mathrm{Res}_p\, (C_1, C_4) = \neg p \vee q \vee r$

$\mathrm{Res}_?\, (C_1, C_5)$ NU

$C_7 = \mathrm{Res}_q\, (C_1, C_6) = \neg p \vee p$

$\mathrm{Res}_?\, (C_2, C_3)$ NU

$\mathrm{Res}_?\, (C_2, C_4)$ NU

$\mathrm{Res}_?\, (C_2, C_5)$ NU

$C_8 = \mathrm{Res}_q\, (C_2, C_6) = \neg p$

$\mathrm{Res}_?\, (C_3, C_4)$ NU

$C_9 = \mathrm{Res}_r\, (C_3, C_5) = q$

$C_{10} = \mathrm{Res}_q\, (C_3, C_6) = r$

$C_2 = \mathrm{Res}_r\, (C_4, C_5) = \neg p \vee q$

$C_{11} = \mathrm{Res}_q\, (C_4, C_6) = r \vee \neg p$

$\mathrm{Res}_?\, (C_5, C_6)$ NU

$S_2^{\,1} = \{C_7 \,,\, C_8 \,,\, C_9 \,,\, C_{10} \,,\, C_{11}\} = \{\neg p \vee p,\, \neg p,\, q,\, r,\, r \vee \neg p \}$

Strategia saturării pe nivele aplicată la S_2 (1):

$S_2 = \{ C_1, C_2, C_3, C_4, C_5, C_6 \} =$
$= \{ \neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r, \neg q \}$

$C_2 = Res_p (C_1, C_2) = \neg p \vee q$

$Res_? (C_1, C_3)$ NU

$C_4 = Res_p (C_1, C_4) = \neg p \vee q \vee r$

$Res_? (C_1, C_5)$ NU

$C_7 = Res_q (C_1, C_6) = \neg p \vee p$

$Res_? (C_2, C_3)$ NU

$Res_? (C_2, C_4)$ NU

$Res_? (C_2, C_5)$ NU

$C_8 = Res_q (C_2, C_6) = \neg p$

$Res_? (C_3, C_4)$ NU

$C_9 = Res_r (C_3, C_5) = q$

$C_{10} = Res_q (C_3, C_6) = r$

$C_2 = Res_r (C_4, C_5) = \neg p \vee q$

$C_{11} = Res_q (C_4, C_6) = r \vee \neg p$

$Res_? (C_5, C_6)$ NU

$S_2^1 = \{ C_7, C_8, C_9, C_{10}, C_{11} \} = \{ \neg p \vee p, \neg p, q, r, r \vee \neg p \}$

Strategia saturării pe nivele aplicată la $S_2(2)$:

$S_2^0 = \{ C_1, C_2, C_3, C_4, C_5, C_6 \} =$

$= \{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r, \neg q\}$

$S_2^1 = \{C_7, C_8, C_9, C_{10}, C_{11}\} =$

$= \{\neg p \vee p, \neg p, q, r, r \vee \neg p\}$

$C_1 = Res_p(C_7, C_1) = \neg p \vee p \vee q$

$C_2 = Res_p(C_7, C_2) = \neg p \vee q$

$Res_?(C_7, C_3)$ NU

$C_4 = Res_p(C_7, C_4) = r \vee \neg p \vee q$

$Res_?(C_7, C_5)$ NU

$Res_?(C_7, C_6)$ NU

$C_8 = Res_p(C_7, C_8) = \neg p$

$Res_?(C_7, C_9)$ NU

$Res_?(C_7, C_{10})$ NU

$C_{11} = Res_p(C_7, C_{11}) = r \vee \neg p$

$C_2 = Res_p(C_8, C_1) = \neg p \vee q$

$Res_?(C_8, C_2)$ NU

$Res_?(C_8, C_3)$ NU

$Res_?(C_8, C_4)$ NU

$Res_?(C_8, C_3)$ NU

$Res_?(C_8, C_4)$ NU

$Res_?(C_8, C_5)$ NU

$Res_?(C_8, C_6)$ NU

$Res_?(C_8, C_9)$ NU

$Res_?(C_8, C_{10})$ NU

$Res_?(C_8, C_{11})$ NU

$Res_?(C_9, C_1)$ NU

$Res_?(C_9, C_2)$ NU

$Res_?(C_9, C_3)$ NU

$Res_?(C_9, C_4)$ NU

$Res_?(C_9, C_5)$ NU

$C_{12} = Res_p(C_9, C_6) = \square$

$\Rightarrow S_2$ este inconsistentă

Strategia mulţimii suport

- se *evită* aplicarea regulii de rezoluţie asupra unor clauze dintr-o *submulţime consistentă* a mulţimii iniţiale de clauze, deoarece rezolvenţii obţinuţi sunt *irelevanţi* în procesul de derivare a □
- Această strategie a fost inspirată din faptul următor: în general mulţimea *premizelor* (faptelor) unei deducţii este *consistentă*, deci rezolvarea unor clauze din această mulţime consistentă nu poate duce la derivarea clauzei vide (inconsistenţa)

- **Definiţie:** Fie S o mulţime de clauze. O submulţime Y a lui S se numeşte *mulţime suport* a lui S, dacă $S \setminus Y$ este consistentă. **Rezoluţia mulţimii suport** este rezoluţia a două clauze care nu aparţin ambele mulţimii $S \setminus Y$.

Aplicarea strategiei mulțimii suport pt. S_2:

$S_2 = \{ C_1, C_2, C_3, C_4, C_5, C_6\} =$

$= \{\neg p \vee p \vee q, \neg p \vee q, q \vee r, r \vee \neg p \vee q, \neg r, \neg q\}$

$Y = \{C_6\}$ mulțimea suport a lui S_2. este formată din clauzele obținute din concluzie.

$S_2 \setminus Y = S_1$ și am demonstrat deja că S_1 este consistentă.

Așadar nu vom rezolva clauzele S_1 din între ele.

Vom începe de la o clauză din Y și nu vom rezolva deloc primele 5 clauze între ele:

$C_7 = Res_r (C_5, C_3) = q_{TCC}$

$C_8 = Res_r (C_6, C_7) = \square \Rightarrow S_2$ e inconsistentă adică, Vasile își face laboratorul

Logică computațională
Curs 7

Lector dr. Pop Andreea-Diana

Rafinările rezoluției

- impun restricții asupra clauzelor care rezolvă, pentru a eficientiza procesul rezolutiv

Notație

- $S \vdash_{\text{Res}}^{st} \square$ ”din mulțimea S de clauze s-a derivat clauza vidă prin aplicarea strategiei *st* a rezoluției propoziționale”

Completitudinea și corectitudinea

- Toate rafinările şi strategiile rezolutive păstrează completitudinea şi corectitudinea.
- Combinarea lor poate impune prea multe restricții și deși mulțimea inițială de clauze este inconsistentă, s-ar putea să nu se poată deriva clauza vidă.
- sunt complete:
  - rezoluţia generală + strategia eliminării
  - rezoluţia generală + strategia mulţimii suport
  - rezoluţia generală + strategia mulţimii suport + strategia eliminării
  - rezoluția liniară + strategia eliminării
  - rezoluţia liniară + strategia mulţimii suport
- nu sunt complete:
  - rezoluția blocării + strategia eliminării
  - rezoluția blocării + strategia mulţimii suport
  - rezoluția blocării + rezoluția liniară
  - rezoluţia unitară
  - rezoluția de intrare

Rezoluţia blocării (lock resolution)

- introdusă de Boyer în 1971

- fiecare apariţie de literal din mulţimea de clauze este indexat arbitrar cu un întreg
- *restricţia:* literalii care rezolvă din clauzele părinţi trebuie să aibă **cei mai mici indici** din aceste clauze
- literalii din rezolvenţi moştenesc indicii de la clauzele părinţi, iar în cazul moştenirii a doi literali identici, se păstrează cel cu indicele mai mic

- este foarte eficientă şi uşor de implementat, se recomandă combinarea ei cu strategia saturării pe nivele

Teorema de corectitudine și completitudine

- Teorema de completitudine

  Fie *S* o mulţime de clauze în care fiecare literal este indexat în mod arbitrar cu un întreg. Dacă *S* este inconsistentă, atunci există o deducţie din mulțimea *S* a clauzei vide prin rezoluţia blocării.

- Teorema de corectitudine

  Fie *S* o mulţime de clauze în care fiecare literal este indexat în mod arbitrar cu un întreg. Dacă din *S* se deduce prin rezoluţia blocării clauza vidă, atunci *S* este inconsistentă.

Exemple (1)

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția blocării:

- $S = \{ \neg r , p \vee \neg q , r \vee p \vee q , \neg q \vee \neg p \}$

Rezolvare

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția blocării:

$$S = \{{}_{(1)}\neg r\ ,\ {}_{(6)}\,p \vee {}_{(2)}\neg q\ ,\ {}_{(8)}\,r \vee {}_{(3)}\,p \vee {}_{(5)}\,q\ ,\ {}_{(4)}\neg q \vee {}_{(7)}\neg p\}$$

$$C_1 \overset{\text{not.}}{\equiv} {}_{(1)}\boldsymbol{\neg r}$$

$$C_2 \overset{\text{not.}}{\equiv} {}_{(2)}\boldsymbol{\neg q} \vee {}_{(6)}\,p$$

$$C_3 \overset{\text{not.}}{\equiv} {}_{(3)}\,\boldsymbol{p} \vee {}_{(5)}\,q \vee {}_{(8)}\,r$$

$$C_4 \overset{\text{not.}}{\equiv} {}_{(4)}\boldsymbol{\neg q} \vee {}_{(7)}\neg p$$

$\xRightarrow{\text{TCC}}$ S este consistentă

Exemple (2)

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția blocării:

- $S = \{ \neg r , p \vee \neg q , r \vee p \vee q , \neg q \vee \neg p \}$
- $S = \{ p \vee q , \neg p \vee q , p \vee \neg q , \neg p \vee \neg q \}$

Rezolvare

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția blocării:

$$S = \{{}_{(5)}p \vee {}_{(4)}\ q\ ,\ {}_{(1)}\neg\, p \vee {}_{(6)}\ q\ ,\ {}_{(7)}p \vee {}_{(2)}\neg\, q\ ,\ {}_{(8)}\neg\, p \vee {}_{(3)}\neg\, q\ \}$$

$C_1 \overset{\text{not.}}{=\!=} {}_{(4)}\, q \vee {}_{(5)}\, p$

$C_2 \overset{\text{not.}}{=\!=} {}_{(1)}\neg\, p \vee {}_{(6)}\, q$

$C_3 \overset{\text{not.}}{=\!=} {}_{(2)}\neg\, q \vee {}_{(7)}\, p$

$C_4 \overset{\text{not.}}{=\!=} {}_{(3)}\neg\, q \vee {}_{(8)}\neg\, p$

$C_5 = \mathrm{Res}_q^{\ lock}\,(C_1, C_3) = {}_{(5)}\, \boldsymbol{p}$

$C_6 = \mathrm{Res}_p^{\ lock}\,(C_2, C_5) = {}_{(6)}\, q$

$C_7 = \mathrm{Res}_q^{\ lock}\,(C_4, C_6) = {}_{(8)}\neg\, \boldsymbol{p}$

$C_8 = \mathrm{Res}_p^{\ lock}\,(C_5, C_7) = \square$

$\overset{\text{TCC}}{\Longrightarrow}$ S este inconsistentă

Observație!!!!!!

$$C_1 \overset{\text{not.}}{=} q \vee p$$
$$C_4 \overset{\text{not.}}{=} \neg q \vee \neg p$$

$$A \vee l,\ B \vee \neg l \vdash_{res} A \vee B$$
$$\mathrm{Res}_p(C_1, C_4) = q \vee \neg q \equiv \mathrm{T}$$
$$\mathrm{Res}_q(C_1, C_4) = p \vee \neg p \equiv \mathrm{T}$$

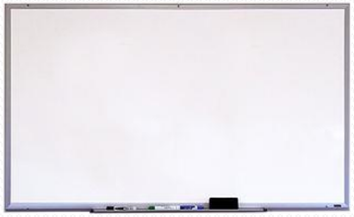
$\square \equiv \mathbf{F}$

$\mathbf{U} \wedge \neg \mathbf{U} \equiv \mathbf{F}$ $\qquad \neg(q \vee p) \equiv \neg q \wedge \neg p \neq \neg q \vee \neg p$

Exerciții (1)

- rezoluția blocării + strategia eliminării nu e completă
  - $S = \{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$

    $C_1 =_{(2)} p \vee_{(1)} q$, $C_2 =_{(3)} \neg p \vee_{(4)} q$, $C_3 =_{(5)} p \vee_{(6)} \neg q$, $C_4 =_{(8)} \neg p \vee_{(7)} \neg q$



Exerciții (1)

- rezoluția blocării + strategia eliminării nu e completă
  - $S = \{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$

    $C_1 =_{(2)} p \vee_{(1)} q,\ C_2 =_{(3)} \neg p \vee_{(4)} q,\ C_3 =_{(5)} p \vee_{(6)} \neg q,\ C_4 =_{(8)} \neg p \vee_{(7)} \neg q$

$\cancel{C_?} = \mathrm{Res}_p^{\,lock}(C_2, C_3) = \cancel{_{(4)}q \vee_{(6)} \neg q}$ conform str. eliminării este o clauză tautologică, deci se va elimina

$\cancel{C_?} = \mathrm{Res}_q^{\,lock}(C_1, C_4) = \cancel{_{(2)}p \vee_{(8)} \neg p}$ conform str. eliminării este o clauză tautologică, deci se va elimina

Nu se mai rezolvă clauze noi, deci nu putem ajunge la □, deci, am ajunge la concluzia greșită că S nu e inconsistentă.

Exerciții (1)

- rezoluția blocării fără strategia eliminării e completă
  - $S = \{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$

    $C_1 = {}_{(2)}p \vee_{(1)} q$, $C_2 = {}_{(3)}\neg p \vee_{(4)} q$, $C_3 = {}_{(5)}p \vee_{(6)} \neg q$, $C_4 = {}_{(8)}\neg p \vee_{(7)} \neg q$

$C_5 = \mathrm{Res}_p^{\,lock}(C_2, C_3) = {}_{(4)}q \vee_{(6)} \neg q$

$C_6 = \mathrm{Res}_q^{\,lock}(C_4, C_5) = {}_{(6)}\neg q \vee_{(8)} \neg p$

$C_8 = \mathrm{Res}_q^{\,lock}(C_6, C_1) = {}_{(2)}p \vee {}_{(8)}\neg p$

$C_9 = \mathrm{Res}_p^{\,lock}(C_8, C_2) = {}_{(4)}q \vee {}_{(8)}\neg p$

$C_{10} = \mathrm{Res}_q^{\,lock}(C_9, C_4) = {}_{(8)}\neg p$

$C_{11} = \mathrm{Res}_p^{\,lock}(C_{10}, C_3) = {}_{(6)}\neg q$ $\xRightarrow{\text{TCC}}$ S este inconsistentă

$C_{12} = \mathrm{Res}_q^{\,lock}(C_{11}, C_1) = {}_{(2)}p$

$C_{13} = \mathrm{Res}_p^{\,lock}(C_{12}, C_{10}) = \square$

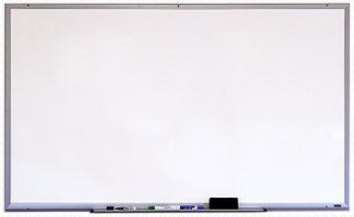
Exerciții (1)

- rezoluția blocării + strategia eliminării nu e completă
  - $S = \{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$

    $C_1 =_{(2)} p \vee_{(1)} q$, $C_2 =_{(3)} \neg p \vee_{(4)} q$, $C_3 =_{(5)} p \vee_{(6)} \neg q$, $C_4 =_{(8)} \neg p \vee_{(7)} \neg q$
- rezoluția blocării + strategia mulţimii suport nu e completă
  - $p \to (q \to r), r \wedge s \to t, u \to s \wedge \neg t \vdash p \wedge q \to \neg u$

    $C_1 =_{(3)} \neg p \vee_{(2)} \neg q \vee_{(1)} r$, $C_2 =_{(6)} \neg r \vee_{(5)} \neg s \vee_{(4)} t$, $C_3 =_{(8)} \neg u \vee_{(7)} s$, $C_4 =_{(10)} \neg u \vee_{(9)} \neg t$, $C_5 =_{(11)} p$, $C_6 =_{(12)} q$, $C_7 =_{(13)} u$

    $Y = \{C_5, C_6, C_7\}$



Exerciții (1)

- rezoluția blocării + strategia mulţimii suport nu e completă
  - $p \to (q \to r), r \wedge s \to t\,, u \to s \wedge \neg\, t \vdash p \wedge q \to \neg\, u$

    $C_1 =_{(3)} \neg p \vee_{(2)} \neg q \vee_{(1)} r$, $C_2 =_{(6)} \neg r \vee_{(5)} \neg s \vee_{(4)} t$, $C_3 =_{(8)} \neg u \vee_{(7)} s$, $C_4 =_{(10)} \neg u \vee_{(9)} \neg t$, $C_5 =_{(11)} p$, $\mathrm{C}_6 =_{(12)} q$, $C_7 =_{(13)} u$

    $Y=\{C_5\,,\,C_6\,,\,C_7\}$

În această indexare, nu rezolvă nici o clauză urmând strategia mulțimii suport, deci am ajunge la falsa concluzie că S este consistentă.

Exerciții (1)

- rezoluția blocării fără strategia mulţimii suport:

$p \to (q \to r), r \wedge s \to t, u \to s \wedge \neg t \vdash p \wedge q \to \neg u$

$C_1 = {}_{(3)}\neg p \vee {}_{(2)}\neg q \vee {}_{(1)}r$, $C_2 = {}_{(6)}\neg r \vee {}_{(5)}\neg s \vee {}_{(4)}t$, $C_3 = {}_{(8)}\neg u \vee {}_{(7)}s$,
$C_4 = {}_{(10)}\neg u \vee {}_{(9)}\neg t$, $C_5 = {}_{(11)}p$, $C_6 = {}_{(12)}q$, $C_7 = {}_{(13)}u$

$C_8 = \mathrm{Res}_t^{\ lock}(C_2, C_4) = {}_{(5)}\neg s \vee {}_{(6)}\neg r \vee {}_{(10)}\neg u$

$C_9 = \mathrm{Res}_s^{\ lock}(C_8, C_3) = {}_{(6)}\neg r \vee {}_{(8)}\neg u$

$C_{10} = \mathrm{Res}_r^{\ lock}(C_1, C_9) = {}_{(2)}\neg q \vee {}_{(3)}\neg p \vee {}_{(8)}\neg u$

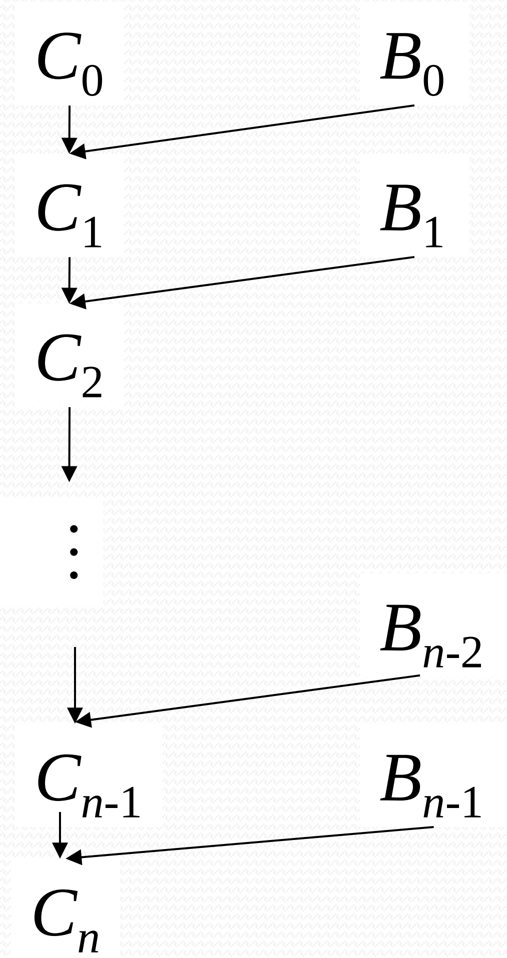
$C_{11} = \mathrm{Res}_q^{\ lock}(C_{10}, C_6) = {}_{(3)}\neg p \vee {}_{(8)}\neg u$

$C_{12} = \mathrm{Res}_p^{\ lock}(C_{11}, C_5) = {}_{(8)}\neg u$ $\xRightarrow{TCC}$ S este inconsistentă

$C_{13} = \mathrm{Res}_u^{\ lock}(C_{12}, C_7) = \square$

Rezoluţia liniară

- Loveland 1970
- procesul rezolutiv este liniar: la fiecare pas una dintre clauzele părinte este rezolventul obţinut la pasul anterior
- Arborele de derivare corespunzător procesului rezolutiv liniar are forma:
  - C_0 clauză vârf
  - C_1, C_2,…, C_n clauze centrale
  - B_0, B_1,…, $B_{n\text{-}1}$ clauze laterale
  - $\forall i$=1,2,…,n, are loc: C_i = Res ($C_{i\text{-}1}$, $B_{i\text{-}1}$)



Teorema de corectitudine și completitudine

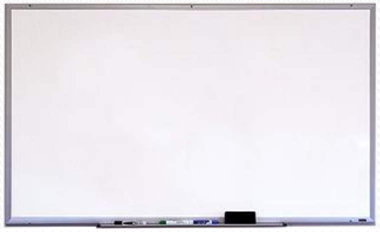
- Mulțimea S de clauze este inconsistentă, dacă și numai dacă $S \vdash_{\text{Res}}^{lin} \square$.

Exerciții (2)

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția liniară:

- $S = \{p \vee q \,,\, \neg\, p \vee q \,,\, p \vee \neg\, q \,,\, \neg\, p \vee \neg\, q\,\}$

$\boxed{C_1 = p \vee q}$ $\boxed{C_2 = \neg\, p \vee q}$ $\boxed{C_3 = p \vee \neg\, q}$ $\boxed{C_4 = \neg\, p \vee \neg\, q}$



$C_1 = p \vee q$ $\quad$ $C_2 = \neg p \vee q$ $\quad$ $C_3 = p \vee \neg q$ $\quad$ $C_4 = \neg p \vee \neg q$

$C_5 = q$ $\quad$ $C_3 = p \vee \neg q$

$C_6 = p$ $\quad$ $C_4 = \neg p \vee \neg q$

$C_7 = \neg q$ $\quad$ $C_5 = q$

$C_8 = \square$

$\xRightarrow{\text{TCC}}$ S este inconsistentă

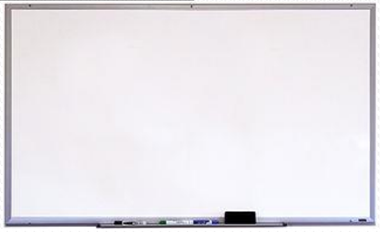
Exerciții (2)

Verificați inconsistența mulțimilor următoare de clauze utilizând rezoluția liniară:

- $S = \{p \vee q \,,\, \neg\, p \vee q \,,\, p \vee \neg\, q \,,\, \neg\, p \vee \neg\, q\,\}$

- $S = \{\neg\, r \,,\, p \vee \neg\, q \,,\, r \vee p \vee q \,,\, \neg\, q \vee \neg\, p\}$ (+strategia eliminării)

$C_1 = \neg\, r$ $C_2 = p \vee \neg\, q$ $C_3 = r \vee p \vee q$ $C_4 = \neg\, q \vee \neg\, p$



$C_1 = \neg r$ | $C_2 = p \vee \neg q$ | $C_3 = r \vee p \vee q$ | $C_4 = \neg q \vee \neg p$

$C_1 = \neg r$ | $C_3 = r \vee p \vee q$

$C_5 = p \vee q$ | $C_2 = p \vee \neg q$

$C_6 = p$ | $C_4 = \neg q \vee \neg p$

$C_7 = \neg q$ | $C_3 = r \vee p \vee q$

$C_4 = \neg q \vee \neg p$ | $C_8 = r \vee p$ | $C_1 = \neg r$

$C_1 = \neg r$ | $C_9 = \neg q \vee r$ | $C_6 = p$

1. Nu îl mai luăm tot pe C_4, nu avem altă opțiune, revenim un pas mai sus

$C_3 = r \vee p \vee q$ | $C_7 = \neg q$ | $C_5 = p \vee q$

$C_8 = r \vee p$ | $C_6 = p$

2. nu avem altă opțiune, revenim mai sus

3. Mai sus...
Începem cu alte 2 clauze care revolvă
Și reîncercăm (backtracking)...
Tot nu ajungem la clauza vidă

$\overset{TCC}{\Longrightarrow}$ *S* este consistentă

Observație:

- rezoluția liniară furnizează o strategie la nivel de implementare: *căutarea cu revenire*
  - la fiecare iterație, pentru clauza centrală pot exista mai multe posibile clauze laterale
  - după ce au fost utilizate toate posibilele clauze laterale, dar nu s-a obținut clauza vidă, se revine la iterația precedentă
  - consistența mulțimii de clauze este demonstrată după o căutare completă fără derivarea clauzei vide

Cazuri particulare ale rezoluției liniare

- **Rezoluția unitară** (*unit*): clauzele centrale au *cel puţin* o *clauză părinte unitară* (conţine un singur literal)
- **Rezoluția de intrare** (*input*): clauzele *laterale* sunt clauze *iniţiale* (de intrare)

Teorema de echivalenţă dintre rezoluţia unit şi cea input

- Fie mulţimea S de clauze. $S \vdash^{input}_{Res} \Box$ dacă şi numai dacă $S \vdash^{unit}_{Res} \Box$.

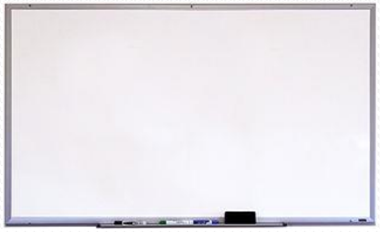
- **corectitudinea**: Dacă $S \vdash^{input/unit}_{Res} \Box$ atunci S este inconsistentă
- **incompletitudinea:** există mulţimi inconsistente de clauze din care nu se poate deriva clauza vidă folosind rezoluţia input sau rezoluţia unit.

Exerciții (3)

Verificați inconsistența mulțimilor următoare de clauze utilizând rafinările input și unit:

- $S = \{ \neg r, p \vee \neg q, r \vee q, \neg q \vee \neg p \}$

$\boxed{C_1 = \neg r}$ $\boxed{C_2 = p \vee \neg q}$ $\boxed{C_3 = r \vee q}$ $\boxed{C_4 = \neg q \vee \neg p}$



Input (cl. lat. sunt din mulț. init.) și unit (C_5,...C_8 au cel puțin un părinte unit.)

$C_1 = \neg r$ $\quad$ $C_2 = p \vee \neg q$ $\quad$ $C_3 = r \vee q$ $\quad$ $C_4 = \neg q \vee \neg p$

$C_1 = \underline{\neg r}$ $\quad$ $C_3 = \underline{r} \vee q$

$C_5 = \underline{q}$ $\quad$ $C_2 = p \vee \underline{\neg q}$

$C_6 = \underline{p}$ $\quad$ $C_4 = \neg q \vee \underline{\neg p}$

$C_7 = \underline{\neg q}$ $\quad$ $C_3 = r \vee \underline{q}$

$C_8 = \underline{r}$ $\quad$ $C_1 = \underline{\neg r}$

$C_9 = \square$

$\xRightarrow{TCC}$ S este inconsistentă

Tipuri de metode

	Semantice	**Sintactice**
Directe	Tabela de adevăr FNC	Deducția (*mp*)
prin **Respingere**	FND Tabele semantică	Rezoluția (generală, strategia eliminării, strategia saturării pe nivele, strategia mulțimii suport, rafinarea rezoluției blocării, rafinarea rezoluției liniare, cazuri particulare: input și unit)