LOGICA MATEMATICĂ ŞI COMPUTAŢIONALĂ
Sem. I, 2017-2018

Ioana Leustean
FMI, UB

Partea III

- **Calculul propozițional clasic**
  - Consistență și satisfiabilitate
  - Teorema de completitudine
  - Algebra Lindenbaum-Tarski
  - Forma normală conjunctivă
  - Clauze
  - Rezoluție
  - Algoritmul Davis-Putnam
  - Demonstrații folosind rezoluția

Consistență, satisfiabilitate, completitudine

Mulţimi consistente

Definiţie

O mulțime Γ de formule se numește **consistentă** dacă există o formulă φ astfel încât $\Gamma \not\vdash \varphi$.

Propoziţie

$\emptyset$ este consistentă.

Dem. Aratăm ča $\not\vdash \neg(\varphi \to \varphi)$. Presupunem că $\vdash \neg(\varphi \to \varphi)$. Deoarece deducția este corectă, obținem $f_e(\neg(\varphi \to \varphi)) = 1$ oricare ar fi e o evaluare. Dar $f_e(\neg(\varphi \to \varphi)) = \neg(f_e(\varphi) \to f_e(\varphi)) = 0$, deci obținem o contradicție.

Exerciţiu. $Teoreme := \{\varphi \in Form \mid\ \vdash \varphi\}$ este consistentă.
Exerciţiu. Orice mulțime satisfiabilă este consistentă.

Mulţimi inconsistente

Definiţie

O mulțime Γ de formule se numeste **inconsistentă** dacă nu este consistentă, i.e. $\Gamma \vdash \varphi$ oricare $\varphi \in Form$.

Propoziţie

Pentru o mulțime $\Gamma \subseteq Form$ sunt echivalente:

(1) Γ este inconsistentă,

(2) există $\psi \in Form$, $\Gamma \vdash \psi$ si $\Gamma \vdash \neg\psi$.

Dem. (1) $\Rightarrow$ (2) este evidentă.
(2) $\Rightarrow$ (1) se demonstrează folosind teorema $\vdash \psi \to (\neg\psi \to \varphi)$.

Exerciţiu. $\Gamma \subseteq \Delta$, Γ inconsistentă $\Rightarrow$ Δ inconsistentă

Mulţimi inconsistente

Propoziţie

Pentru $\Gamma \subseteq Form$ si $\varphi \in Form$ sunt echivalente:

(1) $\Gamma \cup \{\varphi\}$ este inconsistentă,

(2) $\Gamma \vdash \neg\varphi$.

Dem. (1) $\Rightarrow$ (2) $\Gamma \cup \{\varphi\} \vdash \neg\varphi$
$\Gamma \vdash \varphi \to \neg\varphi$ TD
$\Gamma \vdash (\varphi \to \neg\varphi) \to \neg\varphi$ (teoremă)
$\Gamma \vdash \neg\varphi$

(2) $\Rightarrow$ (1) rezultă folosind propoziția anterioară, deoarece $\Gamma \cup \{\varphi\} \vdash \varphi$ si $\Gamma \cup \{\varphi\} \vdash \neg\varphi$.

Mulţimi maximal consistente

Definiţie

O mulțime$\Delta \subset Form$ se numește **maximal consistentă** dacă Δ este consistentă și

$$\text{or. } \Delta' \subseteq Form \ (\Delta \subset \Delta' \Rightarrow \Delta' \text{ inconsistentă}).$$

Propoziţie

Dacă $\Delta \subseteq Form$ este maximal consistentă, atunci sunt adevărate următoarele proprietăți:

- oricare $\varphi \in Form$, $\varphi \in \Delta$ sau $\neg\varphi \in \Delta$,
- oricare φ, $\psi \in Form$
  $\varphi \to \psi \in \Delta \Leftrightarrow \neg\varphi \in \Delta$ sau $\psi \in \Delta$.

Propoziţie

Pentru orice mulțime consistentă $\Gamma \subseteq Form$ este inclusă într-o mulțime maximal consistentă.

Completitudine

Teorema de completitudine extinsă

Orice mulțime consistentă este satisfiabilă.

Dem. Fie $\Gamma \subseteq Form$ o multime consistentă. Trebuie sa demonstrăm că Γ are un model. Fie Δ o mulțimea maximal consistentă a.î. $\Gamma \subseteq \Delta$. Definim $e : Var \to \{0, 1\}$ prin

$$e(v) = 1 \text{ dacă } v \in \Delta,\ e(v) = 0 \text{ dacă } v \notin \Delta.$$

Fie $\boldsymbol{P}(\varphi) =$ " $f_e(\varphi) = 1 \Leftrightarrow \varphi \in \Delta$ ". Demonstrăm prin inducție structurală că $\boldsymbol{P}(\varphi)$ este adevarată pentru orice $\varphi \in Form$. Din definiția evaluării e, $\boldsymbol{P}(v)$ este adevărată oricare $v \in Var$.
Presupunem că $\boldsymbol{P}(\varphi)$ si $\boldsymbol{P}(\psi)$ sunt adevărate. Ob tinem
$f_e(\neg\varphi) = 1 \Leftrightarrow f_e(\varphi) = 0 \Leftrightarrow \varphi \notin \Delta \Leftrightarrow \neg\varphi \in \Delta$,
$f_e(\varphi \to \psi) = f_e(\varphi) \to f_e(\psi) = 1 \Leftrightarrow f_e(\varphi) = 0$ sau
$f_e(\psi) = 1 \Leftrightarrow \varphi \notin \Delta$ sau $\psi \in \Delta \Leftrightarrow \varphi \to \psi \in \Delta$.
Conform principiului inducției structurale, $\boldsymbol{P}(\varphi)$ este adevărată pentru orice $\varphi \in Form$, adică $f_e(\varphi) = 1 \Leftrightarrow \varphi \in \Delta$.
Rezultă $f_e(\Gamma) = \{1\}$, deci Γ admite un model.

Completitudine

Teorema de completitudine

Γ-teoremele și Γ-tautologiile coincid, i.e.

$$\Gamma \vdash \varphi \Leftrightarrow \Gamma \models \varphi$$

oricare are fi $\varphi \in \mathit{Form}$ si $\Gamma \subseteq \mathit{Form}$.
În particular, $\vdash \varphi \Leftrightarrow \models \varphi$.

Dem. $\Rightarrow$ corectitudinea.

$\Leftarrow$ Presupunem că $\Gamma \not\vdash \varphi$. Atunci $\Gamma \not\vdash \neg\neg\varphi$ si $\Gamma \cup \{\neg\varphi\}$ este consistentă. Fie $e : \mathit{Var} \to \{0, 1\}$ un model pentru $\Gamma \cup \{\neg\varphi\}$. Obținem $f_e(\Gamma) = \{1\}$ si $f_e(\neg\varphi) = 1$, deci $f_e(\varphi) = 0$. Din ipoteză, orice model al lui Γ este și model al lui φ, deci $f_e(\varphi) = 1$. Am ajuns la o contradicție, deci presupunerea noastră a fost falsă. În consecință, $\Gamma \vdash \varphi$.

Conectorii derivaţi

Conectori derivaţi

Pentru φ si ψ formule oarecare, definim urmatoarele prescurtari:
$\varphi \vee \psi := \neg\varphi \rightarrow \psi$
$\varphi \wedge \psi := \neg(\varphi \rightarrow \neg\psi)$
$\varphi \leftrightarrow \psi := (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$

Exerciţiu. Dacă $e : Var \rightarrow \{0, 1\}$ evaluare, atunci
$f_e(\varphi \vee \psi) = f_e(\varphi) \vee f_e(\psi)$,
$f_e(\varphi \wedge \psi) = f_e(\varphi) \wedge f_e(\psi)$,
$f_e(\varphi \leftrightarrow \psi) = f_e(\varphi) \leftrightarrow f_e(\psi)$.

Conectori derivaţi

Aratati ca $\vdash v_1 \rightarrow (v_2 \rightarrow (v_1 \wedge v_2))$.

Aplicând teorema de completitudine, aceasta revine la a demonstra $\models v_1 \rightarrow (v_2 \rightarrow (v_1 \wedge v_2))$

v_1	v_2	$v_1 \rightarrow (v_2 \rightarrow (v_1 \wedge v_2))$
0	0	1
0	1	1
1	0	1
1	1	1

Sintaxa şi Semantica

Lemă

Sunt echivalente:

(1) $\vdash \varphi \to \psi$,

(2) $f_e(\varphi) \leq f_e(\psi)$ oricare $e : Var \to \{0, 1\}$ evaluare.

Lemă

Sunt echivalente:

(1) $\vdash \varphi \leftrightarrow \psi$,

(2) $f_e(\varphi) = f_e(\psi)$ oricare $e : Var \to \{0, 1\}$ evaluare.

Dem. exerciţiu

$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Demonstrația sintactică

(1)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash (\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$ (teoremă)
(2)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \varphi \to \psi$ (ipoteză)
(3)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\psi \to \neg\varphi$ (1), (2), MP
(4)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\psi$ (ipoteză)
(5)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\varphi$ (3), (4), MP
(6)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\varphi \to \chi$ (ipoteză)
(7)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \chi$ (5), (6), MP
(8)	$\{\varphi \to \psi, \neg\varphi \to \chi\} \vdash \psi \vee \chi$ TD
(9)	$\{\varphi \to \psi\} \vdash (\varphi \vee \chi) \to (\psi \vee \chi)$ TD
(10)	$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$ TD

Sintaxa şi Semantica

Demonstrația semantică

$\models (v_1 \to v_2) \to ((v_1 \vee v_3) \to (v_2 \vee v_3))$

Metoda tabelului

v_1	v_2	v_3	$(v_1 \to v_2) \to ((v_1 \vee v_3) \to (v_2 \vee v_3))$
0	0	0	1
0	0	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$
1	1	1	1

Lema substituției. Fie γ o formula si $v_1, \ldots, v_n$ variabilele care apar în γ. Fie $\gamma(\gamma_1, \ldots, \gamma_n)$ formula obținută înlocuind v_i cu γ_i oricare $i \in \{1, \ldots, n\}$. Dacă $\models \gamma$ atunci $\models \gamma(\gamma_1, \ldots, \gamma_n)$.
Dem. exercițiu

Rezultă $\models (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Algebra Lindenbaum-Tarski

Echivalenţa formulelor

Form mulțimea formulelor calculului propozițional clasic PC.

Pe *Form* definim relația binară $\sim$ prin

$$\varphi \sim \psi \text{ ddacă } \vdash \varphi \leftrightarrow \psi \text{ ddacă } \vdash \varphi \rightarrow \psi \text{ si } \vdash \psi \rightarrow \varphi,$$

Teoremă

$\sim \subseteq$ *Form* $\times$ *Form* este o relație de echivalență pe *Form*

Dem. Relația $\sim$ este reflexivă, simetrică și tranzitivă deoarece:

$\vdash \varphi \leftrightarrow \varphi$

$\vdash \varphi \leftrightarrow \psi \Leftrightarrow \vdash \psi \leftrightarrow \varphi$

$\vdash \varphi \leftrightarrow \psi$ si $\vdash \psi \leftrightarrow \chi \Rightarrow \vdash \varphi \leftrightarrow \chi$

Algebra Lindenbaum-Tarski

Pe *Form*, mulțimea formulelor PC, definim relația binară $\sim$ prin

$$\varphi \sim \psi \Leftrightarrow \vdash \varphi \leftrightarrow \psi.$$

Pe mulțimea claselor de echivalență $Form/\sim = \{\widehat{\varphi} \mid \varphi \in Form\}$ definim operațiile $\vee$, $\wedge$, $\neg$, 0, 1 prin

$$\hat{\varphi} \vee \hat{\psi} := \widehat{\varphi \vee \psi}, \;\; \hat{\varphi} \wedge \hat{\psi} := \widehat{\varphi \wedge \psi}, \;\; \overline{\hat{\varphi}} := \widehat{\neg\varphi},$$
$$1 := \widehat{\theta},\ 0 := \widehat{(\neg\theta)} \text{ (cu } \theta \text{ teoremă).}$$

În acest fel obținem o **structură algebrică**

$$\text{LINDA} = (Form/\sim,\ \vee, \wedge, ^{-}, 0, 1)$$

Algebra LINDA se numește **algebra Lindenbaum-Tarski** a PC.

LINDA este o **algebră Boole**

Algebra Boole

Definiţie

O **algebră Boole** este o structură

$$(A, \vee, \wedge, ^{-}, 0, 1)$$

care satisface următoarele identităţi or. $x, y, z \in A$:

(L1) $(x \vee y) \vee z = x \vee (y \vee z)$, $(x \wedge y) \wedge z = x \wedge (y \wedge z)$,

(L2) $x \vee y = y \vee x$, $x \wedge y = y \wedge x$,

(L3) $x \vee (x \wedge y) = x$, $x \wedge (x \vee y) = x$,

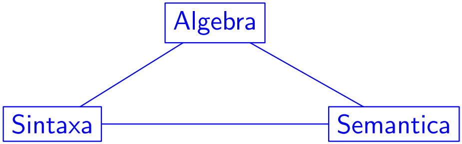
(D) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$, $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$,

(P) $x \vee 0 = x$, $x \wedge 0 = 0$,

(U) $x \wedge 1 = x$, $x \vee 1 = 1$,

(C) $x \vee \overline{x} = 1$, $x \wedge \overline{x} = 0$.

SAS



Teoremă de completitudine

Pentru $\varphi \in \mathit{Form}$, sunt echivalente:

(1) $\vdash \varphi$,

(2) $\models \varphi$ $\quad$ ($f_e(\varphi) = 1$ pentru orice evaluare $e : \mathit{Var} \to L_2$),

(3) $\hat{\varphi} = 1$ în LINDA.

$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Demonstrația sintactică

(1)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash (\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$ (teoremă)
(2)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \varphi \to \psi$ (ipoteză)
(3)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\psi \to \neg\varphi$ (1), (2), MP
(4)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\psi$ (ipoteză)
(5)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\varphi$ (3), (4), MP
(6)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \neg\varphi \to \chi$ (ipoteză)
(7)	$\{\varphi \to \psi, \neg\varphi \to \chi, \neg\psi\} \vdash \chi$ (5), (6), MP
(8)	$\{\varphi \to \psi, \neg\varphi \to \chi\} \vdash \psi \vee \chi$ TD
(9)	$\{\varphi \to \psi\} \vdash (\varphi \vee \chi) \to (\psi \vee \chi)$ TD
(10)	$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$ TD

Demonstrația semantică

$\models (v_1 \to v_2) \to ((v_1 \vee v_3) \to (v_2 \vee v_3))$

Metoda tabelului

v_1	v_2	v_3	$(v_1 \to v_2) \to ((v_1 \vee v_3) \to (v_2 \vee v_3))$
0	0	0	1
0	0	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$
1	1	1	1

Lema substituției. Fie γ o formula si v_1, ..., v_n variabilele care apar în γ. Fie $\gamma(\gamma_1, \ldots, \gamma_n)$ formula obținută înlocuind v_i cu γ_i oricare $i \in \{1, \ldots, n\}$. Dacă $\models \gamma$ atunci $\models \gamma(\gamma_1, \ldots, \gamma_n)$.

Rezultă $\models (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Demonstrația algebrică

$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Dacă $\theta := (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$, atunci
$\hat{\theta} := (\hat{\varphi} \to \hat{\psi}) \to ((\hat{\varphi} \vee \hat{\chi}) \to (\hat{\psi} \vee \hat{\chi}))$ in LINDA .

Notăm $a := \hat{\varphi}$, $b := \hat{\psi}$, $c := \hat{\chi}$ si obținem

$$\begin{aligned}
\hat{\theta} &= (a \to b) \to ((a \vee c) \to (b \vee c)) \\
&= (\overline{a} \vee b) \to ((\overline{a} \wedge \overline{c}) \vee c \vee b) \\
&= (a \wedge \overline{b}) \vee b \vee (\overline{a} \wedge \overline{c}) \vee c \\
&= (a \vee b) \vee (\overline{a} \vee c) = 1
\end{aligned}$$

Am arătat că $\hat{\theta} = 1$ in LINDA, deci $\vdash \theta$.

Demonstrația algebrică

$\vdash (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$
Notăm $\theta := (\varphi \to \psi) \to ((\varphi \vee \chi) \to (\psi \vee \chi))$

Fie B o algebră Boole și $e : Var \to B$ o evaluare.

Notăm $a := f_e(\varphi)$, $b := f_e(\psi)$, $c := f_e(\chi)$ si obținem

$$\begin{aligned}
f_e(\theta) &= (a \to b) \to ((a \vee c) \to (b \vee c)) \\
&= (\overline{a} \vee b) \to ((\overline{a} \wedge \overline{c}) \vee c \vee b) \\
&= (a \wedge \overline{b}) \vee b \vee (\overline{a} \wedge \overline{c}) \vee c \\
&= (a \vee b) \vee (\overline{a} \vee c) = 1_B
\end{aligned}$$

Am arătat că $f_e(\theta) = 1_B$ pentru orice algebră Boole B și orice evaluare $e : Var \to B$, deci $\vdash \theta$.

Forma normală conjunctivă

Exemplu

Arătați că $\models v_1 \to (v_2 \to (v_1 \wedge v_2))$.

v_1	v_2	$v_1 \to (v_2 \to (v_1 \wedge v_2))$
0	0	1
0	1	1
1	0	1
1	1	1

Acest tabel definește o funcție $F : \{0, 1\}^2 \to \{0, 1\}$

x_1	x_2	$F(x_1, x_2)$
0	0	1
0	1	1
1	0	1
1	1	1

Funcţia asociată unei formule

Fie φ o formulă $v_1, \ldots, v_n$ variabilele care apar în φ, $e_1, \ldots, e_{2^n}$ evaluările posibile. Tabelul asociat

v_1	v_2	$\ldots$	v_n	φ
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$e_k(v_1)$	$e_k(v_2)$	$\ldots$	$e_k(v_n)$	$f_{e_k}(\varphi)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

definește funcția $F_\varphi : \{0,1\}^n \to \{0,1\}$

x_1	x_2	$\ldots$	x_n	$F_\varphi(x_1, \ldots, x_n)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$e_k(v_1)$	$e_k(v_2)$	$\ldots$	$e_k(v_n)$	$f_{e_k}(\varphi)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Funcţia asociată unei formule

Fie φ o formulă cu variabilele v_1, ..., v_n.

Definiţie

Funcția asociată lui φ este $F_\varphi : \{0,1\}^n \to \{0,1\}$ definită prin

$$F_\varphi(x_1, \ldots, x_n) = f_e(\varphi),$$

unde $e(v_1) = x_1$, $e(v_2) = x_2$, ..., $e(v_n) = x_n$

Definiţie

O **funcție Booleană** în n variabile este o funcție

$$F : \{0,1\}^n \to \{0,1\}.$$

F_φ este o funcție Booleană pentru orice φ.

Lemă

Sunt echivalente:

(1) $\vdash \varphi \rightarrow \psi$,

(2) $f_e(\varphi) \leq f_e(\psi)$ oricare $e : Var \rightarrow \{0, 1\}$ evaluare,

(3) $F_\varphi \leq F_\psi$.

Lemă

Sunt echivalente:

(1) $\vdash \varphi \leftrightarrow \psi$,

(2) $f_e(\varphi) = f_e(\psi)$ oricare $e : Var \rightarrow \{0, 1\}$ evaluare,

(3) $F_\varphi = F_\psi$.

Dem. exerciţiu

Caracterizarea funcţiilor Booleene

Teorema FB

Pentru orice funcție Booleană $H : \{0,1\}^n \to \{0,1\}$ există o formulă φ cu n variabile astfel încât $H = F_\varphi$.

Exemplu:

Fie $H : \{0,1\}^3 \to \{0,1\}$ descrisă prin tabelul:

x	y	z	$H(x,y,z)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$M_1 = x \vee y \vee z$

$M_2 = x \vee y \vee \neg z$

$m_1 = \neg x \wedge y \wedge \neg z$

$M_3 = x \vee \neg y \vee \neg z$

$m_2 = x \wedge \neg y \wedge \neg z$

$m_3 = x \wedge \neg y \wedge z$

$m_4 = x \wedge y \wedge \neg z$

$m_5 = x \wedge y \wedge z$

Fie $\varphi = M_1 \wedge M_2 \wedge M_3$ și $\psi = m_1 \vee m_2 \vee m_3 \vee m_4 \vee m_5$

$H(x,y,z) = F_\varphi = F_\psi$

FND si FNC

Definiţie

Un literal este o variabilă sau negația unei variabile.

O *formă normală disjunctivă* (FND) este o disjuncție de conjuncții de literali.

O *formă normală conjuctivă* (FNC) este o conjuncție de disjuncții de literali.

Teoremă

Pentru orice formulă φ există o FND θ_1 si o FNC θ_2 astfel încât $\models \varphi \leftrightarrow \theta_1$ si $\models \varphi \leftrightarrow \theta_2$

Dem. Fie θ_1 si θ_2 formulele definite în demonstrația teoremei anterioare pentru $H = F_\varphi$. Atunci $F_\varphi = F_{\theta_1} = F_{\theta_2}$.

FNC

Orice formulă poate fi adusa la FNC prin urmatoarele transformări:

1. înlocuirea implicațiilor și echivalențelor

$$\varphi \to \psi \sim \neg\varphi \vee \psi,$$
$$\varphi \leftrightarrow \psi \sim (\neg\varphi \vee \psi) \wedge (\neg\psi \vee \varphi)$$

2. regulile De Morgan

$$\neg(\varphi \vee \psi) \sim \neg\varphi \wedge \neg\psi,$$
$$\neg(\varphi \wedge \psi) \sim \neg\varphi \vee \neg\psi,$$

3. principiului dublei negații

$$\neg\neg\psi \sim \psi$$

4. distributivitatea

$$\varphi \vee (\psi \wedge \chi) \sim (\varphi \vee \psi) \wedge (\varphi \vee \chi),$$
$$(\psi \wedge \chi) \vee \varphi \sim (\psi \vee \varphi) \wedge (\chi \vee \varphi)$$

Exemple

1. Determinați FNC pentru formula

$(\neg p \rightarrow \neg q) \rightarrow (p \rightarrow q)$
$\sim \neg(\neg p \rightarrow \neg q) \vee (p \rightarrow q)$
$\sim \neg(p \vee \neg q) \vee (\neg p \vee q)$
$\sim (\neg p \wedge q) \vee (\neg p \vee q)$
$\sim (\neg p \vee \neg p \vee q) \wedge (q \vee \neg p \vee q)$

2. Determinați FNC pentru formula

$\neg((p \wedge q) \rightarrow q)$
$\sim \neg(\neg(p \wedge q) \vee q)$
$\sim p \wedge q \wedge \neg q$

Rezoluția în PC

Satisfiabilitate

Definiţie

Fie Γ o mulțime de formule. Un model al lui Γ este o evaluare $e : Var \to \{0, 1\}$ cu $f_e(\Gamma) = \{1\}$. Mulțimea Γ este **satisfiabilă** dacă are un model.
O formulă φ este consecință (semantică) a lui Γ ($\Gamma \models \varphi$) dacă orice model al lui Γ este și model pentru φ.

Teoremă.

Sunt echivalente:

- $\{\varphi_1, \ldots, \varphi_n\} \models \psi$,
- $\{\varphi_1, \ldots, \varphi_n, \neg\psi\}$ nu este satisfiabilă,
- $\varphi_1 \wedge \ldots \wedge \varphi_n \wedge \neg\psi$ nu este satisfiabilă.

Dem. exercițiu

SAT*

Problema satisfiabilităţii: SAT

Fiind dată o formulă φ (în forma normală conjunctivă) există o evaluare $e : Var \to \{0,1\}$ astfel încât $f_e(\varphi) = 1$?

Teorema Cook-Levin

SAT este o problemă NP-completă.

Stephen Cook 1971, Leonid Levin 1973

Cook, Stephen A. (1971). "The Complexity of Theorem-Proving Procedures". Proceedings of the 3rd Annual ACM Symposium on Theory of Computing: 151-158.

Rezoluţia

- Rezoluția propozițională este o regula de deducție pentru calculul propozițional clasic.
- Multe demonstratoare automate și SAT-solvere au la bază rezoluția.
- Utilizând rezoluția se poate construi un demonstrator automat corect și complet pentru calculul propozițional, fara alte teoreme și reguli de deducție.
- Limbajul PROLOG este fundamentat de rezoluție.

Rezoluția este o metodă de verificare a satisfiabilității pentru formule în forma clauzală.

Clauze

Definitii

Un **literal** este o variabila sau negatia unei variabile.
O **clauza** este o multime finita de literali:
$C = \{L_1, \ldots, L_n\}$, unde L_1, ..., L_n sunt literali.
Clauza C este **triviala** daca exista $p \in Var$ astfel incat p, $\neg p \in C$.

O clauza $C = \{L_1, \ldots, L_n\}$ este satisfabila daca formula $L_1 \vee \ldots \vee L_n$ este satisfiabila. Daca $e : Var \to \{0, 1\}$ este o evaluare vom nota $e(C) := f_e(L_1 \vee \ldots \vee L_n)$.

Clauza vida $\square := \{\}$ nu este satisfiabila (disjunctie indexata de $\emptyset$).

O multime de clauze $\mathcal{S} = \{C_1, \ldots, C_m\}$ este satisfiabila daca exista o evaluare $e : Var \to \{0, 1\}$ astfel incat $e(C_i) = 1$ oricare $i \in \{1, \ldots, m\}$.

Clauze

Observatie
Putem identifica
clauza $C = \{L_1, \ldots, L_n\}$ cu $L_1 \vee \ldots \vee L_n$,
multimea de clauze $\mathcal{S} = \{C_1, \ldots, C_m\}$ cu $C_1 \wedge \ldots \wedge C_m$.

clauza $\approx$ disjunctie de literali

multime de clauze $\approx$ FNC

Definitie
C clauza, $\mathcal{S}$ multime de clauze
$Var(C) = \{p \in Var | p \in C \text{ sau } \neg p \in C\}$,
$Var(\mathcal{S}) = \bigcup\{Var(C) | C \in \mathcal{S}\}$.

Exemple

1. p, $\neg r$, q sunt literali.

2. $\{p, \neg r\}$, $\{\neg r, r\}$, $\{q, p\}$, $\{q, \neg p, r\}$ sunt clauze.

3. $\mathcal{S} = \{\{p, \neg r\}, \{\neg r, r\}, \{q, p\}, \{q, \neg p, r\}\}$ este satisfiabila.
Dem. Consideram $e(p) = e(q) = 1$.

4. $\mathcal{S} = \{\{\neg p, q\}, \{\neg r, \neg q\}, \{p\}, \{r\}\}$ nu este satisfiabila.
Dem. Daca exista o evaluare e care satisface $\mathcal{C}$, atunci $e(p) = e(r) = 1$. Rezulta $e(q) = 0$, deci $f_e(\{\neg p, q\}) = f_e(\neg p \vee q) = 0$. In consecinta, $\{\neg p, q\}$ nu e satisfacuta de e.

5. O multime de clauze triviale este intotdeauna satisfiabila.

Proprietati

Propozitie

Fie C, D clauze si $\mathcal{T}$, $\mathcal{S}$, $\mathcal{U}$ multimi de clauze. Urmatoarele implicatii sunt adevarate.

(p1) $C \subseteq D$, C satisfiabila $\Rightarrow$ D satisfiabila

(p2) $C \cup D$ satisfiabila $\Rightarrow$ C satisfiabila sau D satisfiabila

(p3) $p \notin Var(C) \Rightarrow C \cup \{p\}$ și $C \cup \{\neg p\}$ sunt satisfiabile

(p4) $\mathcal{S} \subseteq \mathcal{T}$, $\mathcal{T}$ satisfiabila $\Rightarrow$ $\mathcal{S}$ satisfiabila

(p5) Fie $p \in Var$ si $\mathcal{U}$, $\mathcal{T}$, $\mathcal{S}$ multimi de clauze astfel incat

$$\begin{gathered} p \notin Var(\mathcal{U}), \\ \text{or. } T \in \mathcal{T}\ (p \in T \text{ si } \neg p \notin T), \\ \text{or. } S \in \mathcal{S}\ (p \notin S \text{ si } \neg p \in S). \end{gathered}$$

Atunci $\mathcal{U}$ satisfiabila $\Rightarrow \mathcal{U} \cup \mathcal{T}$, $\mathcal{U} \cup \mathcal{S}$ satisfiabile

Dem. (p1)-(p4) exercitiu

Regula Rezolutiei

$$Rez \quad \frac{C_1 \cup \{p\}, C_2 \cup \{\neg p\}}{C_1 \cup C_2}$$

unde $\{p, \neg p\} \cap C_1 = \emptyset$ si $\{p, \neg p\} \cap C_2 = \emptyset$.

Propozitie

Regula Rezolutiei pastreaza satsifiabilitatea, i.e.
$\{C_1 \cup \{p\}, C_2 \cup \{\neg p\}\}$ satisfiabila $\Leftrightarrow$ $C_1 \cup C_2$ satisfiabila.

Dem. Fie $e : Var \to \{0, 1\}$ este o evaluare astfel incat $e(C_1 \cup \{p\}) = e(C_2 \cup \{\neg p\}) = 1$.Daca $e(p) = 1$ atunci $e(C_2) = 1$, deci $e(C_1 \cup C_2) = 1$. Similar pentru $e(p) = 0$.
Invers, presupunem ca $e(C_1 \cup C_2) = 1$, deci $e(C_1) = 1$ sau $e(C_2) = 1$. Daca $e(C_1) = 1$ consideram $e' : Var \to \{0, 1\}$ o evaluare cu $e'(v) = e(v)$ daca $v \in Var(C_1)$ si $e'(p) = 0$. Atunci $e'(C_1) = e(C_1) = 1$, deci $e'(C_1 \cup \{p\}) = 1$.De asemenea $e'(C_2 \cup \{\neg p\}) = e'(C_2) \vee e'(\neg p) = 1$, deci e' este model pentru multimea de clauze $\{C_1 \cup \{p\}, C_2 \cup \{\neg p\}\}$.

Exemple

$$\frac{\{p\}, \{\neg p\}}{\square} \quad \frac{\{p\}, \{\neg p, q\}}{\{q\}}$$

Atentie

Aplicarea **simultana** a regulii pentru doua variabile diferit este gresita. De exemplu

$$\frac{\{p, \neg q\}, \{\neg p, q\}}{\square}$$

contrazice rezultatul anterior, deoarece $\{\{p, \neg q\}, \{\neg p, q\}\}$ este satisfiabila ($e(p) = e(q) = 1$).
Aplicarea corectă a Regulii Rezolutiei este

$$\frac{\{p, \neg q\}, \{\neg p, q\}}{\{q, \neg q\}}$$

.

Derivare prin rezolutie

Definitie

Fie $\mathcal{S}$ o multime de clauze. O *derivare prin rezolutie* din $\mathcal{S}$ este o secventa finita de clauze astfel incat fiecare clauza este din $\mathcal{S}$ sau rezulta din clauze anterioare prin rezolutie.

Exemplu $\mathcal{S} = \{\{\neg p, q\}, \{\neg q, \neg r, s\}, \{p\}, \{r\}, \{\neg s\}\}$

O derivare prin rezolutie pentru $\square$ din $\mathcal{S}$ este

$C_1 = \{\neg s\}$ $(\in \mathcal{S})$
$C_2 = \{\neg q, \neg r, s\}$ $(\in \mathcal{S})$
$C_3 = \{\neg q, \neg r\}$ $(C_1,\ C_2,\ Rez)$
$C_4 = \{r\}$ $(\in \mathcal{S})$
$C_5 = \{\neg q\}$ $(C_3,\ C_4,\ Rez)$
$C_6 = \{\neg p, q\}$ $(\in \mathcal{S})$
$C_7 = \{\neg p\}$ $(C_5,\ C_6,\ Rez)$
$C_8 = \{p\}$ $(\in \mathcal{S})$
$C_9 = \square$ $(C_7,\ C_8,\ Rez)$

Algoritmul Davis-Putnam(DP)

Verifică satisfiabilitatea unei mulţimi de clauze

Intrare: $\mathcal{S}$ multime nevida de clauze netriviale.

$\mathcal{S}_1 := \mathcal{S}$; $i := 1$;

P1. $v_i \in Var(\mathcal{C}_i)$;

$\mathcal{T}_i^1 := \{C \in \mathcal{S}_i | v_i \in C\}$;

$\mathcal{T}_i^0 := \{C \in \mathcal{S}_i | \neg v_i \in C\}$;

$\mathcal{T}_i := \mathcal{T}_i^0 \cup \mathcal{T}_i^1$;

$\mathcal{U}_i := \emptyset$;

P2. if $\mathcal{T}_i^0 \neq \emptyset$ and $\mathcal{T}_i^1 \neq \emptyset$ then

$\mathcal{U}_i := \{C_1 \setminus \{v_i\} \cup C_0 \setminus \{\neg v_i\} | C_1 \in \mathcal{T}_i^1, C_0 \in \mathcal{T}_i^0\}$;

P3. $\mathcal{S}_{i+1} = (\mathcal{S}_i \setminus \mathcal{T}_i) \cup \mathcal{U}_i$; $\mathcal{S}_{i+1} = \mathcal{S}_{i+1} \setminus \{C \in \mathcal{S}_{i+1} | C \text{ triviala}\}$;

P4. if $\mathcal{S}_{i+1} = \emptyset$ then *SAT*

else if $\square \in \mathcal{S}_{i+1}$ then *NESAT*

else $\{i := i + 1$; go to P1$\}$.

Run DP

Exemplu

$\mathcal{S}_1 := \mathcal{S} = \{\{\neg p, q, \neg s\}, \{\neg r, \neg q\}, \{p, r\}, \{p\}, \{r\}, \{s\}\}$

$v_1 := p;\ T_1^1 := \{\{p, r\}, \{p\}\};\ T_1^0 := \{\{\neg p, q, \neg s\}\};$
$U_1 := \{\{r, q, \neg s\}, \{q, \neg s\}\};$
$\mathcal{S}_2 := \{\{\neg r, \neg q\}, \{r\}, \{s\}, \{r, q, \neg s\}, \{q, \neg s\}\};\ i := 2;$

$v_2 := q;\ T_2^1 := \{\{r, q, \neg s\}, \{q, \neg s\}\};\ T_2^0 := \{\{\neg r, \neg q\}\};$
$U_2 := \{\{r, q, \neg r\}, \{\neg s, \neg r\}\};$
$\mathcal{S}_3 := \{\{r\}, \{s\}, \{\neg s, \neg r\}\};\ i := 3;$

$v_3 := r;\ T_3^1 := \{\{r\}\};\ T_3^0 := \{\{\neg s, \neg r\}\};\ U_3 := \{\{\neg s\}\};$
$\mathcal{S}_4 := \{\{s\}, \{\neg s\}\};\ i := 4;$

$v_4 := s;\ T_4^1 := \{\{s\}\};\ T_4^0 := \{\{\neg s\}\};\ U_4 := \{\square\};$
$\mathcal{S}_5 := \{\square\}$
In consecinta, $\mathcal{S}$ nu este satisfiabila

Exemplu

$\mathcal{S}_1 := \mathcal{S} = \{\{p, \neg r\}, \{q, p\}, \{q, \neg p, r\}\}$

$v_1 := r$; $T_1^1 := \{\{q, \neg p, r\}\}$; $T_1^0 := \{\{p, \neg r\}\}$;
$U_1 := \{\{q, \neg p, p\}\}$;
$\mathcal{S}_2 := \{\{q, p\}\}$; $i := 2$;

$v_2 := q$; $T_2^1 := \{\{q, p\}\}$; $T_2^0 := \emptyset$;
$U_2 := \emptyset$;
$\mathcal{S}_3 := \emptyset$
In consecinta, $\mathcal{S}$ este satisfiabila

Algoritmul DP

Fie $\mathcal{S}$ o multime finita de clauze si n este numarul de variabile care apar in $\mathcal{S}$. Algoritmul DP se opreste deoarece $Var(\mathcal{S}_{i+1}) \subset Var(\mathcal{S}_i)$. Daca n este numarul de variabile care apar in $\mathcal{S}$, atunci exista un $n_0 \leq n$ astfel incat $Var(\mathcal{S}_{n_0+1}) = \emptyset$, deci $\mathcal{S}_{n_0+1} = \emptyset$ sau $\mathcal{S}_{n_0+1} = \{\square\}$.
Pentru simplitate, vom presupune in continuare ca $n_0 = n$.

Propozitie

$\mathcal{S}_i$ satisfiabila $\Leftrightarrow$ $\mathcal{S}_{i+1}$ satisfiabila
oricare $i \in \{1, \ldots, n-1\}$.

Dem. Fie $i \in \{1, \ldots, n-1\}$. Stim ca $\mathcal{S}_{i+1} = (\mathcal{S}_i \setminus \mathcal{T}_i) \cup \mathcal{U}_i$.
Consideram cazurile $U_i = \emptyset$ si $U_i \neq \emptyset$.
Daca $U_i = \emptyset$, atunci $\mathcal{T}_i^0 = \emptyset$ sau $\mathcal{T}_i^1 = \emptyset$ si $\mathcal{S}_i = (\mathcal{S}_{i+1} \cup \mathcal{T}_i)$. Ne aflam in ipotezele proprietatii (p5) . Din (p4) si (p5) obtinem concluzia dorita.

Algoritmul DP

Dem. continuare

Daca $U_i \neq \emptyset$, notam $\mathcal{C}_i := \mathcal{S}_i \setminus \mathcal{T}_i$. In consecinta $\mathcal{S}_i = \mathcal{C}_i \cup \mathcal{T}_i$ si $\mathcal{S}_{i+1} = \mathcal{C}_i \cup \mathcal{U}_i$.Observam ca fiecare clauza din $\mathcal{U}_i$ se obtine aplicand Regula Rezolutiei pentru doua clauze din $\mathcal{T}_i$. Am demonstrat ca Regula Rezolutiei pastreaza satisfiabilitatea:

$$\mathcal{U}_i \text{ satisfiabila} \Leftrightarrow \mathcal{T}_i \text{ satisfiabila.}$$

Au loc urmatoarele echivalente:
$\mathcal{S}_i = \mathcal{C}_i \cup \mathcal{T}_i$ satisfiabila $\Leftrightarrow \mathcal{C}_i$, $\mathcal{T}_i$ satisfiabile
$\Leftrightarrow \mathcal{C}_i, \mathcal{U}_i$ satisfiabile $\Leftrightarrow \mathcal{U}_i \cup \mathcal{T}_i = \mathcal{S}_{i+1}$ satisfiabila.

Algoritmul DP

Teorema DP

Algoritmul DP este corect si complet, i.e. $\mathcal{S}$ nu este satisfiabila daca si numai daca iesirea algoritmului DP este $\{\square\}$.

Dem. Din propozitia anterioara, $\mathcal{S}$ nu este satisfiabila daca si numai daca $\mathcal{S}_n$ nu este satisfiabila. Fie p unica variabila propozitionala care apare in $\mathcal{S}_n$.
Daca $\mathcal{S}_n = \{\{p\}\}$ atunci $\mathcal{S}_n$ este satisfiabila deoarece $e(p) = 1$ este un model. Daca $\mathcal{S}_n = \{\{\neg p\}\}$ atunci $\mathcal{S}_n$ este satisfiabila deoarece $e(p) = 0$ este un model. In ambele cazuri $\mathcal{S}_{n+1} = \{\}$.

Daca $\mathcal{S}_n = \{\{p\}, \square\}$ sau $\mathcal{S}_n = \{\{\neg p\}, \square\}$ atunci $\mathcal{S}_n$ nu este satisfiabila si $\mathcal{S}_{n+1} = \{\square\}$. Daca $\mathcal{S}_n = \{\{p\}, \{\neg p\}\}$ sau $\mathcal{S}_n = \{\{p\}, \{\neg p\}, \square\}$ atunci $\mathcal{S}_n$ nu este satisfiabila, putem aplica Regula Rezolutiei si obtinem $\mathcal{S}_{n+1} = \{\square\}$.

Se observa ca S_n nu este satisfiabila daca si numai daca $S_{n+1} = \{\square\}$.

Rezolutia

Observatie
Algoritmul DP cu intrarea $\mathcal{S}$ se termina cu $\{\square\}$ daca si numai daca exista o derivare prin rezolutie a clauzei vide $\square$ din $\mathcal{S}$.

Teorema
Daca $\mathcal{S}$ este o multime finita nevida de clauze, sunt echivalente:

- $\mathcal{S}$ nu este satisfiabila,
- exista o derivare prin rezolutie a clauzei vide $\square$ din $\mathcal{S}$.

Regula Rezolutiei este corecta si completa pentru PC.

Exemplu

Cercetăm satisfiabilitatea mulțimii
$\mathcal{S} = \{\{\neg p, q, \neg s\}, \{\neg r, \neg q\}, \{p, r\}, \{p\}, \{r\}, \{s\}\}$

$\mathcal{S}_1 := \mathcal{S}$
$v_1 := p;\ T_1^1 := \{\{p, r\}, \{p\}\};\ T_1^0 := \{\{\neg p, q, \neg s\}\};$
$U_1 := \{\{r, q, \neg s\}, \{q, \neg s\}\};$

$C_1 = \{p, r\}\ (\in S)$
$C_2 = \{\neg p, q, \neg s\}\ (\in S)$
$C_3 = \{r, q, \neg s\}\ (C_1, C_2, Rez)$
$C_4 = \{p\}\ (\in S)$
$C_5 = \{q, \neg s\}\ (C_4, C_2, Rez)$

Exemplu

$\mathcal{S}_2 := \{\{\neg r, \neg q\}, \{r\}, \{s\}, \{r, q, \neg s\}, \{q, \neg s\}\}$; $i := 2$;

$v_2 := q$; $T_2^1 := \{\{r, q, \neg s\}, \{q, \neg s\}\}$; $T_2^0 := \{\{\neg r, \neg q\}\}$;
$U_2 := \{\{\neg s, \neg r\}\}$; $\mathcal{S}_3 := \{\{r\}, \{s\}, \{\neg s, \neg r\}\}$; $i := 3$;

$C_6 = \{\neg r, \neg q\}$ $(\in S)$
$C_7 = \{\neg r, r, \neg s\}$ (C_3, C_6, Rez)
$C_8 = \{\neg r, \neg s\}$ (C_5, C_6, Rez)

Exemplu

$\mathcal{S}_3 := \{\{r\}, \{s\}, \{\neg s, \neg r\}\}$; $i := 3$;

$v_3 := r$; $T_3^1 := \{\{r\}\}$; $T_3^0 := \{\{\neg s, \neg r\}\}$; $U_3 := \{\{\neg s\}\}$;

$C_9 = \{r\}$ $(\in S)$
$C_{10} = \{\neg s\}$ (C_9, C_8, Rez)

Exemplu

$\mathcal{S}_4 := \{\{s\}, \{\neg s\}\}$; $i := 4$;

$v_4 := s$; $T_4^1 := \{\{s\}\}$; $T_4^0 := \{\{\neg s\}\}$; $U_4 := \{\square\}$;

$C_{11} = \{s\}$ $(\in S)$
$C_{12} = \{\neg s\}$ (C_{10}, C_{11}, Rez)
$C_{13} = \square$ (C_{11}, C_{12}, Rez)

$\mathcal{S}_5 := \{\square\}$
In consecinta, $\mathcal{S}$ nu este satisfiabila

Forma clauzala

Definitie

Fie φ o formula si $C_1 \wedge \cdots \wedge C_n$ o FNC astfel incat $\models \varphi \leftrightarrow C_1 \wedge \cdots \wedge C_n$. Spunem ca multimea de clauze $\{C_1, \cdots, C_n\}$ este **forma clauzala** a lui φ.

Lema

φ satisfiabila $\Leftrightarrow$ $C_1 \wedge \cdots \wedge C_n$ satisfiabila $\Leftrightarrow \Leftrightarrow \{C_1, \cdots, C_n\}$ satisfiabila

Dem. exercitiu

Definitie

Daca $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ este o multime de formule atunci o forma clauzala pentru Γ este $\mathcal{S} := \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n$ unde $\mathcal{S}_i$ este forma clauzala pentru γ_i oricare i.

Lema

Γ satisfiabila $\Leftrightarrow$ $\mathcal{S}$ satisfiabila

Dem. exercitiu

Demonstratii prin rezolutie

Teorema

Fie Γ o multime finita de formule, φ o formula si $\mathcal{S}$ o forma clauzala pentru $\Gamma \cup \{\neg\varphi\}$. Sunt echivalente:

(1) $\Gamma \models \varphi$,

(2) $\Gamma \cup \{\neg\varphi\}$ nu e satisfiabila,

(3) $\mathcal{S}$ nu este satisfiabila,

(4) exista o derivare prin rezolutie a clauzei vide $\square$ din $\mathcal{S}$.

Observatie

Teorema este adevarata si pentru Γ multime oarecare.
Demonstratia foloseste *compacitatea* calculului propozitional:
o multime de formule este satisfiabila daca si numai daca orice submultime finita a sa este satisfiabila.

Exemplu

A demonstra ca $\models p \to (q \to p)$ revine la a demonstra ca $\{\neg(p \to (q \to p))\}$ nu e satisfiabila.Pentru aceasta determinam o forma clauzala pentru $\neg(p \to (q \to p))$ si aplicam DP.

Determinam FNC pentru $\neg(p \to (q \to p))$:
$\neg(p \to (q \to p)) \sim \neg(\neg p \vee \neg q \vee p) \sim p \wedge q \wedge \neg p$
Forma clauzala este $\mathcal{S} = \{\{p\}, \{q\}, \{\neg p\}\}$.

Aplicam DP:
$\{\{p\}, \{q\}, \{\neg p\}\}$
$\{\{p\}, \{\neg p\}\}$
$\{\square\}$

$\mathcal{S}$ nu este satisfiabila deoarece exista o derivare prin rezolutie a clauzei vide din $\mathcal{S}$. In consecinta, $\models p \to (q \to p)$.

Exemplu

Cercetati daca $\{p \vee q\} \models p \wedge q$. Aceasta revine cerceta satisfiabilitatea multimii $\Delta = \{p \vee q, \neg(p \wedge q)\}$.

O forma clauzala pentru $p \vee q$ este $\{\{p, q\}\}$. O forma clauzala pentru $\neg(p \wedge q)$ este $\{\{\neg p, \neg q\}\}$.
Forma clauzala pentru Δ este $\mathcal{S} = \{\{p, q\}, \{\neg p, \neg q\}\}$.

Aplicam DP:
$\{\{p, q\}, \{\neg p, \neg q\}\}$
$\{\{q, \neg q\}\}$
$\{\}$ (multimea vida)

In acest caz $\mathcal{S}$ este satisfiabila, deci $\{p \vee q\} \not\models p \wedge q$.

Exemplu

Cercetati daca $\{p, p \to (q \vee r)\} \models \neg p \to (\neg p \wedge q \wedge \neg r)$.
Determinam forma clauzala pentru
$\{p, p \to (q \vee r), \neg(\neg p \to (\neg p \wedge q \wedge \neg r))\}$.

Forma clauzala a lui p este $\{\{p\}\}$.

$p \to (q \vee r) \sim \neg p \vee q \vee r$ (FNC)
Forma clauzala a lui $p \to (q \vee r)$ este $\{\{\neg p, q, r\}\}$.

$\neg(\neg p \to (\neg p \wedge q \wedge \neg r)) \sim \neg(p \vee (\neg p \wedge q \wedge \neg r)) \sim \neg p \wedge (p \vee \neg q \vee r))$
Forma clauzala a lui $\neg(\neg p \to (\neg p \wedge q \wedge \neg r))$ este
$\{\{\neg p\}, \{p, \neg q, r\}\}$.

Aplicam DP:
$\{\{p\}, \{\neg p, q, r\}, \{\neg p\}, \{p, \neg q, r\}\}$
$\{\{q, r\}, \square, \{\neg q, r\}\}$
$\{\{r\}, \square\}$
$\{\square\}$
Este adevarat ca $\{p, p \to (q \vee r)\} \models \neg p \to (\neg p \wedge q \wedge \neg r)$

Concluzie

Fie Γ o multime de formule si φ o formula.
Notam prin $\Gamma \vdash_{Rez} \varphi$ faptul ca exista o derivare prin rezolutie a clauzei vide $\square$ dintr-o forma clauzala a lui $\Gamma \cup \{\neg\varphi\}$. In acest caz spunem ca φ se demostreaza prin rezolutie din Γ.

Teorema

Sunt echivalente:

- $\Gamma \models \varphi$,
- $\Gamma \vdash_{Rez} \varphi$.

Folosind rezolutia (fara alte axiome si reguli de deductie) se poate construi un demonstrator automat pentru PC.