Clauze și CNF

Unei mulțimi de clauze $\mathcal{S}$ îi asociem o formulă $\varphi_{\mathcal{S}}$ în FNC astfel:

- $C = \{L_1, \ldots, L_n\}, n \geq 1 \longmapsto \varphi_C := L_1 \vee L_2 \vee \ldots \vee L_n$.
- $\square \longmapsto \varphi_\square := v_0 \wedge \neg v_0$.

Fie $\mathcal{S} = \{C_1, \ldots, C_m\}$ o mulțime nevidă de clauze. Formula asociată lui $\mathcal{S}$ este

$$\varphi_{\mathcal{S}} := \bigwedge_{i=1}^{m} \varphi_{C_i}.$$

Formula asociată mulțimii vide de clauze este $\varphi_\emptyset := v_0 \vee \neg v_0$.
Formula $\varphi_{\mathcal{S}}$ nu este unic determinată, depinde de ordinea în care se scriu elementele în clauze și în $\mathcal{S}$, dar se observă imediat că: $\mathcal{S} = \mathcal{S}'$ implică $\varphi_{\mathcal{S}} \sim \varphi_{\mathcal{S}'}$.

Propoziția 1.88

Pentru orice evaluare $e : V \to \{0, 1\}$, $e \vDash \mathcal{S}$ ddacă $e \vDash \varphi_{\mathcal{S}}$.

Dem.: Exercițiu.

Rezoluția

Definiția 1.89

Fie C_1, C_2 două clauze. O clauză R se numește rezolvent al clauzelor C_1, C_2 dacă există un literal L a.î. $L \in C_1$, $L^c \in C_2$ și

$$R = (C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\}).$$

Regula Rezoluției

$$Rez \quad \frac{C_1, C_2}{(C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\})}, \quad L \in C_1, L^c \in C_2$$

Notăm cu $Res(C_1, C_2)$ mulțimea rezolvenților clauzelor C_1, C_2.

- Rezoluția a fost introdusă de Blake (1937) și dezvoltată de Davis, Putnam (1960) și Robinson (1965).
- Multe demonstratoare automate de teoreme folosesc rezoluția. Limbajul PROLOG este bazat pe rezoluție.

Rezoluția

Exemplu

$C_1 = \{v_1, v_2, \neg v_5\}$, $C_2 = \{v_1, \neg v_2, v_{100}, v_5\}$.

- Luăm $L := \neg v_5$. Atunci $L \in C_1$ și $L^c = v_5 \in C_2$. Prin urmare, $R = \{v_1, v_2, \neg v_2, v_{100}\}$ este rezolvent al clauzelor C_1, C_2.
- Dacă luăm $L' := v_2$, atunci $L' \in C_1$ și $L'^c = \neg v_2 \in C_2$. Prin urmare, $R' = \{v_1, \neg v_5, v_{100}, v_5\}$ este rezolvent al clauzelor C_1, C_2.

Exemplu

$C_1 = \{v_7\}$, $C_2 = \{\neg v_7\}$. Atunci clauza vidă $\square$ este rezolvent al clauzelor C_1, C_2.

Rezoluție

Fie $\mathcal{S}$ o mulțime de clauze.

Definiția 1.90

O derivare prin rezoluție din $\mathcal{S}$ sau o $\mathcal{S}$-derivare prin rezoluție este o secvență $C_1, C_2, \ldots, C_n$ de clauze a.î. pentru fiecare $i \in \{1, \ldots, n\}$, una din următoarele condiții este satisfăcută:

(i) C_i este o clauză din $\mathcal{S}$;

(ii) există $j, k < i$ a.î. C_i este rezolvent al clauzelor C_j, C_k.

Definiția 1.91

Fie C o clauză. O derivare prin rezoluție a lui C din $\mathcal{S}$ este o $\mathcal{S}$-derivare prin rezoluție $C_1, C_2, \ldots, C_n$ a.î. $C_n = C$.

Rezoluție

Exemplu

Fie

$$\mathcal{S} = \{\{\neg v_1, v_2\}, \{\neg v_2, \neg v_3, v_4\}, \{v_1\}, \{v_3\}, \{\neg v_4\}\}.$$

O derivare prin rezoluție a clauzei vide $\Box$ din $\mathcal{S}$ este următoarea:

$$\begin{array}{lcll}
C_1 & = & \{\neg v_4\} & C_1 \in \mathcal{S} \\
C_2 & = & \{\neg v_2, \neg v_3, v_4\} & C_2 \in \mathcal{S} \\
C_3 & = & \{\neg v_2, \neg v_3\} & C_3 \text{ rezolvent al clauzelor } C_1, C_2 \\
C_4 & = & \{v_3\} & C_4 \in \mathcal{S} \\
C_5 & = & \{\neg v_2\} & C_5 \text{ rezolvent al clauzelor } C_3, C_4 \\
C_6 & = & \{\neg v_1, v_2\} & C_6 \in \mathcal{S} \\
C_7 & = & \{\neg v_1\} & C_7 \text{ rezolvent al clauzelor } C_5, C_6 \\
C_8 & = & \{v_1\} & C_8 \in \mathcal{S} \\
C_9 & = & \Box & C_9 \text{ rezolvent al clauzelor } C_7, C_8.
\end{array}$$

Rezoluția

Pentru orice mulțime de clauze $\mathcal{S}$, notăm cu

$$Res(\mathcal{S}) := \bigcup_{C_1, C_2 \in \mathcal{S}} Res(C_1, C_2).$$

Propoziția 1.92

Pentru orice mulțime de clauze $\mathcal{S}$ și orice evaluare $e : V \to \{0, 1\}$,

$$e \models \mathcal{S} \quad \Rightarrow \quad e \models Res(\mathcal{S}).$$

Dem.: Dacă $Res(\mathcal{S}) = \emptyset$, atunci este validă, deci $e \models Res(\mathcal{S})$. Presupunem că $Res(\mathcal{S})$ este nevidă și fie $R \in Res(\mathcal{S})$. Atunci există clauze $C_1, C_2 \in \mathcal{S}$ și un literal L a.î. $L \in C_1, L^c \in C_2$ și $R = (C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\})$. Avem două cazuri:

- $e \models L$. Atunci $e \not\models L^c$. Deoarece $e \models C_2$, există $U \in C_2, U \neq L^c$ a.î. $e \models U$. Deoarece $U \in R$, obținem că $e \models R$.
- $e \models L^c$. Atunci $e \not\models L$. Deoarece $e \models C_1$, există $U \in C_1, U \neq L$ a.î. $e \models U$. Deoarece $U \in R$, obținem că $e \models R$. $\Box$

Corectitudinea rezoluției

Teorema de corectitudine a rezoluției 1.93

Fie $\mathcal{S}$ o mulțime de clauze. Dacă $\Box$ se derivează prin rezoluție din $\mathcal{S}$, atunci $\mathcal{S}$ este nesatisfiabilă.

Dem.: Fie $C_1, C_2, \ldots, C_n = \Box$ o $\mathcal{S}$-derivare prin rezoluție a lui $\Box$.
Presupunem că $\mathcal{S}$ este satisfiabilă și fie $e \models \mathcal{S}$.
Demonstrăm prin inducție după i că:

$$\text{pentru orice } 1 \leq i \leq n,\ e \models C_i.$$

Pentru $i = n$, obținem că $e \models \Box$, ceea ce este o contradicție.
Cazul $i = 1$ este evident, deoarece $C_1 \in \mathcal{S}$.
Presupunem că $e \models C_j$ pentru orice $j < i$. Avem două cazuri:

- $C_i \in \mathcal{S}$. Atunci $e \models C_i$.
- există $j, k < i$ a.î. $C_i \in Res(C_j, C_k)$. Deoarece, conform ipotezei de inducție, $e \models \{C_j, C_k\}$ aplicăm Propoziția 1.92 pentru a conclude că $e \models C_i$.

$\Box$

Algoritmul Davis-Putnam (DP)

Intrare: $\mathcal{S}$ mulțime nevidă de clauze netriviale.
$i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

Pi.1 Fie x_i o variabilă care apare în $\mathcal{S}_i$. Definim

$$\mathcal{T}_i^1 := \{C \in \mathcal{S}_i \mid x_i \in C\}, \quad \mathcal{T}_i^0 := \{C \in \mathcal{S}_i \mid \neg x_i \in C\}.$$

Pi.2 **if** $(\mathcal{T}_i^1 \neq \emptyset$ și $\mathcal{T}_i^0 \neq \emptyset)$ **then**

$$\mathcal{U}_i := \{(C_1 \setminus \{x_i\}) \cup (C_0 \setminus \{\neg x_i\}) \mid C_1 \in \mathcal{T}_i^1, C_0 \in \mathcal{T}_i^0\}.$$

else $\mathcal{U}_i := \emptyset$.

Pi.3 Definim

$$\begin{array}{lcl}
\mathcal{S}'_{i+1} & := & (\mathcal{S}_i \setminus (\mathcal{T}_i^0 \cup \mathcal{T}_i^1)) \cup \mathcal{U}_i; \\
\mathcal{S}_{i+1} & := & \mathcal{S}'_{i+1} \setminus \{C \in \mathcal{S}'_{i+1} \mid C \text{ trivială}\}.
\end{array}$$

Pi.4 **if** $\mathcal{S}_{i+1} = \emptyset$ **then** $\mathcal{S}$ este satisfiabilă.
else if $\Box \in \mathcal{S}_{i+1}$ **then** $\mathcal{S}$ este nesatisfiabilă.
else $\{i := i + 1$; go to Pi.1$\}$.

Algoritmul Davis-Putnam (DP)

$\mathcal{S} = \{\{v_1, \neg v_3\}, \{v_2, v_1\}, \{v_2, \neg v_1, v_3\}\}$. $i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

P1.1 $x_1 := v_3$; $\mathcal{T}_1^1 := \{\{v_2, \neg v_1, v_3\}\}$; $\mathcal{T}_1^0 := \{\{v_1, \neg v_3\}\}$.
P1.2 $\mathcal{U}_1 := \{\{v_2, \neg v_1, v_1\}\}$.
P1.3 $\mathcal{S}_2' := \{\{v_2, v_1\}, \{v_2, \neg v_1, v_1\}\}$; $\mathcal{S}_2 := \{\{v_2, v_1\}\}$.
P1.4 $i := 2$ and go to P2.1.

P2.1 $x_2 := v_2$; $\mathcal{T}_2^1 := \{\{v_2, v_1\}\}$; $\mathcal{T}_2^0 := \emptyset$.
P2.2 $\mathcal{U}_2 := \emptyset$.
P2.3 $\mathcal{S}_3 := \emptyset$.
P2.4 $\mathcal{S}$ este satisfiabilă.

Algoritmul Davis-Putnam (DP)

$\mathcal{S} = \{\{\neg v_1, v_2, \neg v_4\}, \{\neg v_3, \neg v_2\}, \{v_1, v_3\}, \{v_1\}, \{v_3\}, \{v_4\}\}$.
$i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

P1.1 $x_1 := v_1$; $\mathcal{T}_1^1 := \{\{v_1, v_3\}, \{v_1\}\}$; $\mathcal{T}_1^0 := \{\{\neg v_1, v_2, \neg v_4\}\}$.
P1.2 $\mathcal{U}_1 := \{\{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$.
P1.3 $\mathcal{S}_2 := \{\{\neg v_3, \neg v_2\}, \{v_3\}, \{v_4\}, \{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$.
P1.4 $i := 2$ and go to P2.1.

P2.1. $x_2 := v_2$; $\mathcal{T}_2^1 := \{\{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$; $\mathcal{T}_2^0 := \{\{\neg v_3, \neg v_2\}\}$.
P2.2 $\mathcal{U}_2 := \{\{v_3, \neg v_4, \neg v_3\}, \{\neg v_4, \neg v_3\}\}$.
P2.3 $\mathcal{S}_3 := \{\{v_3\}, \{v_4\}, \{\neg v_4, \neg v_3\}\}$.
P2.4 $i := 3$ and go to P3.1.

P3.1 $x_3 := v_3$; $\mathcal{T}_3^1 := \{\{v_3\}\}$; $\mathcal{T}_3^0 := \{\{\neg v_4, \neg v_3\}\}$.
P3.2. $\mathcal{U}_3 := \{\{\neg v_4\}\}$. P3.3 $\mathcal{S}_4 := \{\{v_4\}, \{\neg v_4\}\}$.
P3.4 $i := 4$ and go to P4.1.

P4.1 $x_4 := v_4$; $\mathcal{T}_4^1 := \{\{v_4\}\}$; $\mathcal{T}_4^0 := \{\{\neg v_4\}\}$.
P4.2 $\mathcal{U}_4 := \{\square\}$. P4.3 $\mathcal{S}_5 := \{\square\}$.
P4.4 $\mathcal{S}$ nu este satisfiabilă.

Algoritmul DP - terminare

Notăm:

$Var(C) := \{x \in V \mid x \in C \text{ sau } \neg x \in C\}$, $Var(\mathcal{S}) := \bigcup_{C \in \mathcal{S}} Var(C)$.

Așadar, $Var(C) = \emptyset$ ddacă $C = \square$ și $Var(\mathcal{S}) = \emptyset$ ddacă $\mathcal{S} = \emptyset$ sau $\mathcal{S} = \{\square\}$.

Propoziția 1.94

Fie $n := |Var(\mathcal{S})|$. Atunci algoritmul DP se termină după cel mult n pași.

Dem.: Se observă imediat că pentru orice i,

$$Var(\mathcal{S}_{i+1}) \subseteq Var(\mathcal{S}_i) \setminus \{x_i\} \subsetneq Var(\mathcal{S}_i).$$

Prin urmare, $n = |Var(\mathcal{S}_1)| > |Var(\mathcal{S}_2)| > |Var(\mathcal{S}_3)| > \ldots \geq 0$. $\square$

Fie $N \leq n$ numărul de pași după care se termină DP. Atunci $\mathcal{S}_{N+1} = \emptyset$ sau $\square \in S_{N+1}$.

Algoritmul DP - corectitudine și completitudine

Propoziția 1.95

Pentru orice $i \leq N$,

$$\mathcal{S}_{i+1} \text{ este satisfiabilă} \iff \mathcal{S}_i \text{ este satisfiabilă.}$$

Dem.:
"$\Leftarrow$" Presupunem că $\mathcal{S}_i$ este satisfiabilă și fie $e \vDash \mathcal{S}_i$. Se observă imediat că $\mathcal{S}_{i+1} \subseteq \mathcal{S}_i \cup Res(\mathcal{S}_i)$. Prin urmare, folosind corectitudinea rezoluției, obținem că $e \vDash \mathcal{S}_{i+1}$.
"$\Rightarrow$" Presupunem că $\mathcal{S}_{i+1}$ este satisfiabilă și fie $e \vDash \mathcal{S}_{i+1}$. Deoarece orice clauză trivială este validă, rezultă că $e \vDash \mathcal{S}_{i+1}'$.
Avem următoarele cazuri:

- $\mathcal{T}_i^1 = \emptyset$. Atunci $\mathcal{U}_i = \emptyset$ și $\mathcal{S}_{i+1}' = \mathcal{S}_i \setminus \mathcal{T}_i^0$, deci $\mathcal{S}_i = \mathcal{S}_{i+1}' \cup \mathcal{T}_i^0$. Fie $e' := e_{x_i \leftarrow 0}$. Atunci $e'(x_i) = 0$, deci $e' \vDash \neg x_i$. Rezultă că e' este model pentru orice clauză din $\mathcal{T}_i^0$, adică $e' \vDash \mathcal{T}_i^0$. De asemenea, $e(v) = e'(v)$ pentru orice $v \in Var(\mathcal{S}_{i+1}')$, deci $e' \vDash \mathcal{S}_{i+1}'$. Am obținut că $e' \vDash \mathcal{S}_i$.

- $\mathcal{T}_i^0 = \emptyset$. Se demonstrează similar, folosind evaluarea $e'' := e_{x_i \leftarrow 1}$.
- $\mathcal{T}_i^1 \neq \emptyset$ și $\mathcal{T}_i^0 \neq \emptyset$. Se observă că $\mathcal{S}_i \subseteq \mathcal{S}'_{i+1} \cup (\mathcal{T}_i^1 \cup \mathcal{T}_i^0)$.
  Cazul 1: $e(x_i) = 1$. Definim $e^* := e_{x_i \leftarrow 0}$. Atunci $e, e^* \vDash \mathcal{S}'_{i+1}$, $e \vDash \mathcal{T}_i^1, e^* \vDash \mathcal{T}_i^0$.
  Presupunem că $e, e^* \not\vDash \mathcal{T}_i^1 \cup \mathcal{T}_i^0$. Atunci există $C_1 \in \mathcal{T}_i^1$ a.î. $e^* \not\vDash C_1$ și $C_0 \in \mathcal{T}_i^0$ a.î. $e \not\vDash C_0$. Obținem că $e \not\vDash C_0 \setminus \{\neg x_i\}$.
  Dacă am avea că $e \vDash C_1 \setminus \{x_i\}$, atunci ar exista un literal L care nu conține variabila x_i a.î. $e \vDash L$, de unde am obține că $e^* \vDash L$, contradicție cu faptul că $e^* \not\vDash C_1$.
  Rezultă că $e \not\vDash (C_1 \setminus \{x_i\}) \cup (C_0 \setminus \{\neg x_i\}) \in \mathcal{U}_i \subseteq S'_{i+1}$, o contradicție cu ipoteza.
  Așadar, una din evaluările e, e^* satisface $\mathcal{T}_i^1 \cup \mathcal{T}_i^0$, deci este model pentru $\mathcal{S}_i$.
  Cazul 2: $e(x_i) = 0$. Demonstrația e similară. $\square$

Teorema 1.96

Algoritmul DP este corect și complet, adică,

$$\mathcal{S} \text{ este nesatisfiabilă ddacă } \square \in S_{N+1}.$$

Dem.: Aplicăm Propoziția 1.95. Obținem că $\mathcal{S} = \mathcal{S}_1$ este nesatisfiabilă ddacă S_{N+1} este nesatisfiabilă ddacă $\square \in S_{N+1}$. $\square$