Logică pentru Informatică - Săptămâna 7
Algoritmul lui Tseitin

1 Introducere

Exemplul 1.1. *Să considerăm clasa de formule*

$$\varphi_n = (\mathtt{p_1} \land \mathtt{q_1}) \lor (\mathtt{p_2} \land \mathtt{q_2}) \lor \ldots \lor (\mathtt{p_n} \land \mathtt{q_n}),$$

unde n este un număr natural.

Folosind algoritmul din cursul anterior, obținem următoarea formă normal conjunctivă φ'_n a formulei φ:

$$\varphi'_n = \bigwedge_{a_i \in \{p,q\}} (a_1 \lor a_2 \lor \ldots \lor a_n).$$

Cu alte cuvinte, φ'_n este o conjuncție de 2^n clauze. Acesta este un exemplu de creștere exponențial, și creșterile exponențiale sunt de evitat. De exemplu, dacă $n = 10$, avem că φ'_{10} are 1024 clauze, dacă $n = 20$ avem că φ'_{20} are 1048576 clauze etc. Cu alte cuvinte, dacă n crește cu o unitate, formula rezultată se dublează în dimensiune.

Din păcate, din cauza creșterii exponențiale explicată în exemplul anterior, algoritmul de aducere în FNC nu este deosebit de eficient.

2 Transformarea lui Tseitin

Pentru a evita creșterea exponențială, Tseitin (scris și Tseytin) a propus un algoritm care este mai eficient. Transformarea lui Tseitin (sau algoritmul lui Tseitin) primește la intrare o formulă φ și calculează o formulă φ' care este în FNC. Totuși, spre deosebire de transformarea anterioară, există o legătură mai slabă între φ și φ' în general: este doar garantat că φ și φ' sunt echisatisfiabile, nu neapărat echivalente. Două formule sunt echisatisfiabile dacă ambele sunt satisfiable sau ambele sunt nesatisfiabile.

Exemplul 2.1. *Formulele* $\mathtt{p}$ *și* $\neg\mathtt{q}$ *sunt echisatisfiabile (ambele sunt satisfiabile).*

Formulele $\mathtt{p} \land \neg\mathtt{p}$ *și* $\neg(\mathtt{q} \lor \neg\mathtt{q})$ *sunt echisatisfiabile (ambele sunt nesatisfiabile).*

Formulele $\mathtt{p}$ *și* $\mathtt{p} \land \neg\mathtt{p}$ *nu sunt echisatisfiabile (prima este satisfiabilă, dar a doua e nesatisfiabilă).*

Observația 2.1. *Orice două formule echivalente sunt și echisatisfiabile (justificați).*

Garanțiile oferite de transformarea lui Tseitin sunt prezentate în următoarea teoremă:

Teorema 2.1. *Pentru orice formulă $\varphi \in LP$, există o formulă $\varphi' \in LP$ astfel încât:*

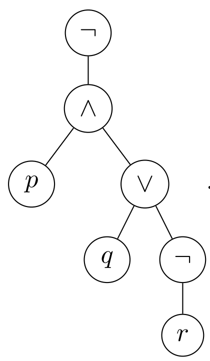
- *φ și φ' sunt echisatisfiabile;*
- *φ' este în CNF;*
- *$size\Big(\varphi'\Big) \leq 16 \times size\Big(\varphi\Big)$ (formula rezultat nu este mult mai mare decât formula de la intrare);*
- *orice model al formulei φ' este de asemenea model al formulei φ.*

Amintiți-vă că prin $size\Big(\varphi\Big)$ desemnam numărul de noduri din arborele de sintaxă abstractă al formulei φ și că funcția *size* a fost definită în cursul 2.

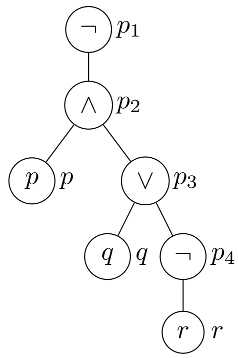
Vom prezenta algoritmul lui Tseitin pe un exemplu.

Pasul 1 Primul pas din algoritmul lui Tseitin este să începem cu arborele abstract al formulei φ și să asociem fiecărui nod o variabilă propozițională.

Fie $\varphi = \neg(\mathtt{p} \wedge (\mathtt{q} \vee \neg \mathtt{r}))$. Avem că

$arb\Big(\varphi\Big) =$  .

Fiecărui nod intern îi asociem o variabilă propozițională *proaspătă* (engl. *fresh*), adică o variabilă propozițională nouă, pe care nu am întâlnit-o până acum. Frunzelor le asociem variabilele caresunt deja în frunze. Iată arborele rezultat, unde lângă fiecare nod scriem variabila propozițională asociată:

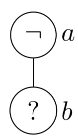


.

Idea din spatele variabilelor propoziționale $\mathtt{p}_1, \mathtt{p}_2, \mathtt{p}_3, \mathtt{p}_4, \mathtt{p}_5$ este să construim o formulă φ' în așa un fel încât în fiecare asigurare τ care satisface φ', valoarea variabilei propoziționale să fie aceeași cu subformula care are rădăcina în nodul asociat variabilei. De exemplu, $\hat{\tau}\Big(p_3\Big)$ ar trebui să fie egal cu $\hat{\tau}\Big(q \vee \neg r\Big)$.

Pasul 2 Formula φ' va fi o conjuncție de clauze și pentru fiecare nod intern vom avea două sau trei clauze, după cum urmează:

- dacă nodul este o negație de forma:

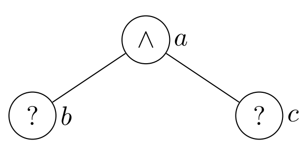
  ,

  unde $a, b \in A$ sunt variabilele propoziționale asociate cu nodurile, atunci adăugăm la φ' clauze echivalente cu formula $a \leftrightarrow \neg b$. În acest fel, în orice atribuire care satisface φ', formulele a și $\neg b$ au aceeași valoare de adevăr. Cum găsim clauzele? Este suficient să rearanjăm formula $a \leftrightarrow \neg b$ după cum urmează:

$$\begin{aligned} \mathtt{a} \leftrightarrow \neg \mathtt{b} &\equiv (a \rightarrow \neg b) \wedge (\neg b \rightarrow a) \\ &\equiv (\neg a \vee \neg b) \wedge (\neg\neg b \vee a) \\ &\equiv (\neg a \vee \neg b) \wedge (b \vee a). \end{aligned}$$

  De aceea, vom adăuga clauzele $\neg a \vee \neg b$ și $b \vee a$ la φ'.

- dacă nodul este o conjuncție de următoarea formă:

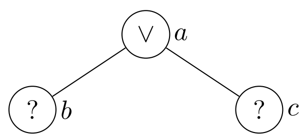
  ,

unde $a, b, c \in A$ sunt variabilele propoziționale asociate nodurilor, atunci adaugăm la φ' clauzele echivalente cu formula $a \leftrightarrow b \wedge c$. În acest fel, în orice atribuire care satisface φ', vom avea că a și $b \wedge c$ au aceeași valoarea de adevăr. Cum găsim clauzele? Este suficient să rearanjăm formula $a \leftrightarrow b \wedge c$ după cum urmează:

$$\begin{aligned} a \leftrightarrow b \wedge c &\equiv (a \rightarrow b \wedge c) \wedge (b \wedge c \rightarrow a) \\ &\equiv (\neg a \vee b \wedge c) \wedge (\neg(b \wedge c) \vee a) \\ &\equiv (\neg a \vee b) \wedge (\neg a \vee c) \wedge (\neg b \vee \neg c \vee a). \end{aligned}$$

De aceea, vom adăuga clauzele $(\neg a \vee b)$, $(\neg a \vee c)$ și $(\neg b \vee \neg c \vee a)$ la φ'.

- dacă nodul este o disjuncție de următoarea formă:



,

unde $a, b, c \in A$ sunt variabilele propoziționale asociate nodurilor, atunci adăguăm la φ' clauzele echivalente cu formula $a \leftrightarrow b \vee c$. În acest fel, în orice atribuire care satisface φ', avem că a și $b \vee c$ au aceeași valoare de adevăr. Cum găsim clauzele? Este suficient să rearanjăm formulă $a \leftrightarrow b \vee c$ după cum urmează:

$$\begin{aligned} a \leftrightarrow b \vee c &\equiv (a \rightarrow b \vee c) \wedge (b \vee c \rightarrow a) \\ &\equiv (\neg a \vee b \vee c) \wedge (\neg(b \vee c) \vee a) \\ &\equiv (\neg a \vee b \vee c) \wedge ((\neg b \wedge \neg c) \vee a) \\ &\equiv (\neg a \vee b \vee c) \wedge (\neg b \vee a) \wedge (\neg c \vee a). \end{aligned}$$

De aceea, vom adăuga clauzele $(\neg a \vee b \vee c)$, $(\neg b \vee a)$ și $(\neg c \vee a)$ la φ'.

- dacă nodul este o implicație, exercițiu pentru acasă;
- dacă nodul este o dublă implicație, exercițiu pentru acasă.

În exemplul nostru, adăugăm următoarele clauze la φ':

1. $\mathtt{p}_1 \vee \mathtt{p}_2$, $\neg\mathtt{p}_1 \vee \neg\mathtt{p}_2$;
2. $\neg\mathtt{p}_2 \vee \mathtt{p}_1$, $\neg\mathtt{p}_2 \vee \mathtt{p}_3$, $\neg\mathtt{p} \vee \neg\mathtt{p}_3 \vee \mathtt{p}_2$;
3. $\neg\mathtt{q} \vee \mathtt{p}_3$, $\neg\mathtt{p}_4 \vee \mathtt{p}_3$, $\neg\mathtt{p}_3 \vee \mathtt{q} \vee \mathtt{p}_4$;
4. $\mathtt{p}_4 \vee \mathtt{r}$, $\neg\mathtt{p}_4 \vee \neg\mathtt{r}$.

Pasul 3 La sfârșit, adăugăm o clauză la φ' care conține doar variabila propozițională asociată rădăcinii.

În exemplul nostru, obținem

$$\begin{array}{rc} \varphi' = & (\mathtt{p}_1 \vee \mathtt{p}_2) \wedge (\neg \mathtt{p}_1 \vee \neg \mathtt{p}_2) \wedge \\ & (\neg \mathtt{p}_2 \vee \mathtt{p}_1) \wedge (\neg \mathtt{p}_2 \vee \mathtt{p}_3) \wedge (\neg \mathtt{p} \vee \neg \mathtt{p}_3 \vee \mathtt{p}_2) \wedge \\ & (\neg \mathtt{q} \vee \mathtt{p}_3) \wedge (\neg \mathtt{p}_4 \vee \mathtt{p}_3) \wedge (\neg \mathtt{p}_3 \vee \mathtt{q} \vee \mathtt{p}_4) \wedge \\ & (\mathtt{p}_4 \vee \mathtt{r}) \wedge (\neg \mathtt{p}_4 \vee \neg \mathtt{r}) \wedge \\ & (p_1) \end{array}$$

Prin construcție, avem că:

1. orice model al formulei φ' este de asemenea un model al formulei φ – deci dacă φ' este satisfiabilă, atunci și φ este satisfiabilă;

2. orice model τ al formulei φ poate fi schimbat într-un model al formulei φ alegând $\tau[\mathtt{p}_i]$ (presupunând că $\mathtt{p}_i$ sunt variabilele noi) ca fiind valoarea de adevăr a subformulei φ corespunzătoare variabilei $\mathtt{p}_i$ în τ – deci dacă φ este satisfiabilă, atunci φ' este de asemenea satisfiabilă;

3. pentru orice nod intern al formulei φ, φ' conine 2 sau 3 clauses, fiecare având un arbore cu cel mult 16 noduri – acest lucru înseamnă că formula rezultată φ' are o dimensiune de cel mult 16 mai mare decât dimensiunea formulei inițiale;

4. din moment ce φ' este o conjuncție de clauze, este în FNC.

Deci φ' satisface toate cerințele din teorema de mai sus.