Základy složitosti a vyčíslitelnosti

NTIN090

Petr Kučera

2021/22 (9. přednáška)

NP-úplnost a důkaz Co-okovy-Levinovy věty

NP-úplnost

Definice

- Jazyk B je NP-těžký, pokud každý jazyk A v NP je polynomiálně převoditelný na B
- Jazyk B je NP-úplný, pokud je NP-těžký a navíc patří do třídy NP

Jak ukazovat NP-úplnost

Věta

Je-li jazyk B NP-úplný a $B \leq_m^P C$ pro nějaký jazyk C v NP, pak jazyk C je také NP-úplný.

Důkaz.

- Z NP-úplnosti B platí pro každý jazyk $A \in \mathrm{NP}$, že $A \leq_m^P B \leq_m^P C$.
- Z tranzitivity $\leq_m^P$ plyne NP-těžkost C
- C je v NP dle předpokladu, je tedy NP-úplný

$\square$

Abychom mohli použít tuto větu, musíme již mít nějaký NP-úplný problém.

První NP-úplný problém

SPLNITELNOST (SAT)

Instance:	Formule φ v KNF.
Otázka:	Je formule φ splnitelná?

Věta

SAT je NP-úplný problém.

- Jako důsledek dostáváme Cookovu-Levinovu větu

Věta (Cookova-Levinova věta)

SAT patří do P, právě když $\mathrm{P} = \mathrm{NP}$.

SAT patří do NP

Lemma

SAT patří do NP

Důkaz.

Polynomiální verifikátor $V(\varphi, \mathbf{a})$ pro SAT:

- Pro danou KNF φ a ohodnocení $\mathbf{a}$
- ověří, zda $\mathbf{a}$ splňuje φ

$\square$

Potřebujeme ještě ukázat, že SAT je NP-těžký.

NP-těžkost SAT

- Nechť A je jazyk v NP
- Popíšeme polynomiální převod A na SAT
- Se vstupem w zkonstruuje převod KNF φ, pro kterou platí

$$w \in A \iff \varphi \text{ je splnitelná}$$

- Formule φ simuluje polynomiální NTS M pro A se vstupem w
- Modely φ popisují přijímající výpočty $M(w)$
  - Pokud $M(w)$ přijme, pak φ má model, který popisuje přijímající výpočet
  - Pokud $M(w)$ nepřijímá, pak φ nemá žádný model

Přijímající výpočet

- Nechť $M = (Q, \Sigma, \delta, q_0, F)$ je nedeterministický TS, který
  - přijímá A (tedy $A = L(M)$)
  - pracuje v čase n^k pro nějaké $k \in \mathbb{N}$
- Vstup w je přijat M, pokud existuje posloupnost konfigurací

$$C_0^w \xrightarrow{\delta} C_1 \xrightarrow{\delta} C_2 \xrightarrow{\delta} \cdots \xrightarrow{\delta} C_{n^k}$$

  - C_0^w je počáteční konfigurace výpočtu $M(w)$
  - Jedna z konfigurací v posloupnosti je přijímající

Technické detaily

- Pro konstrukci bychom měli předpokládat, že M pracuje v čase $n^k - 3$
- Předpokládáme, že přijímající konfigurace může být pomocí δ ponechána beze změny

Tableau

tableau T pro $M(w)$ je matice typu $n^k \times n^k$

- Řádky popisují konfigurace
- Konfigurace začínají a končí znakem $\#$
- Stav je zapsán před políčkem, které je pod hlavou M
- Konfigurace na řádku $i > 1$ následuje z konfigurace na řádku $i - 1$ pomocí přechodové funkce δ

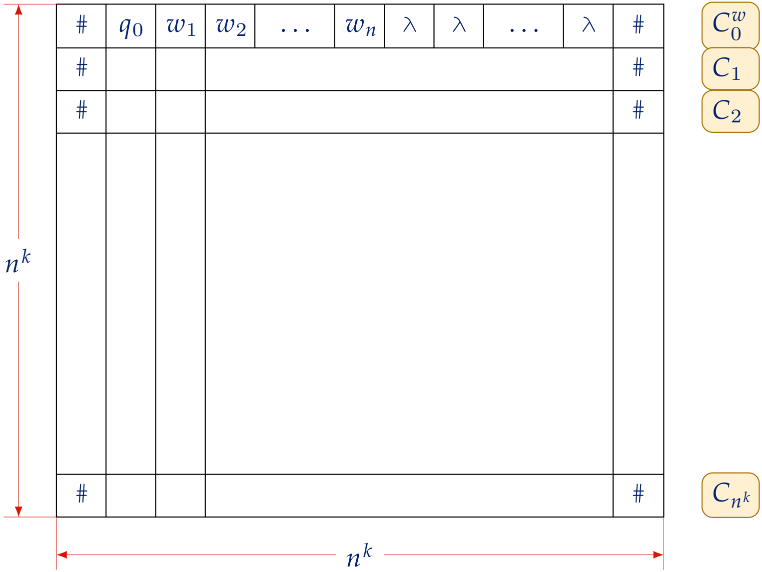
přijímající tableau na nějakém řádku je přijímající konfigurace

buňka jedno políčko tableau

$T[i, j]$ buňka na indexech $i, j \in \{1, \dots, n^k\}$

> M přijímá w, právě když existuje přijímající tableau pro $M(w)$.

Tableau



Proměnné

- $S = Q \cup \Sigma \cup \{\#\}$
  - Množina symbolů použitých v buňkách tableau
- φ má proměnné $x_{i,j,s}$ pro $i,j = 1, \ldots, n^k$ a $s \in S$

> $x_{i,j,s} = 1$ znamená, že $T[i,j]$ obsahuje symbol s

Struktura φ

φ je konjunkcí čtyř podformulí

Každé buňce je přiřazen právě jeden symbol

Každá řádka následuje z předchozí podle přechodové funkce δ

$$\varphi = \varphi_{\text{cell}} \wedge \varphi_{\text{start}} \wedge \varphi_{\text{move}} \wedge \varphi_{\text{accept}}$$

První řádek obsahuje počáteční konfiguraci C_0^w

Jeden z řádků obsahuje přijímající konfiguraci

Sémantika: Každé buňce je přiřazen právě jeden symbol z S

Alespoň jeden symbol v buňce $T[i,j]$

$$\varphi_{\text{cell}} = \bigwedge_{1\le i,j\le n^k} \left[\left(\bigvee_{s\in S} x_{i,j,s} \right) \wedge \left(\bigwedge_{\substack{s,t\in S\\ s\neq t}} (\neg x_{i,j,s} \vee \neg x_{i,j,t}) \right) \right]$$

Nejvýš jeden symbol v buňce $T[i,j]$

Sémantika: První řádek obsahuje počáteční konfiguraci se vstupem w

#	q_0	w_1	w_2	$\dots$	w_n	λ	λ	$\dots$	λ	#

C_0^w

Zakódován pomocí

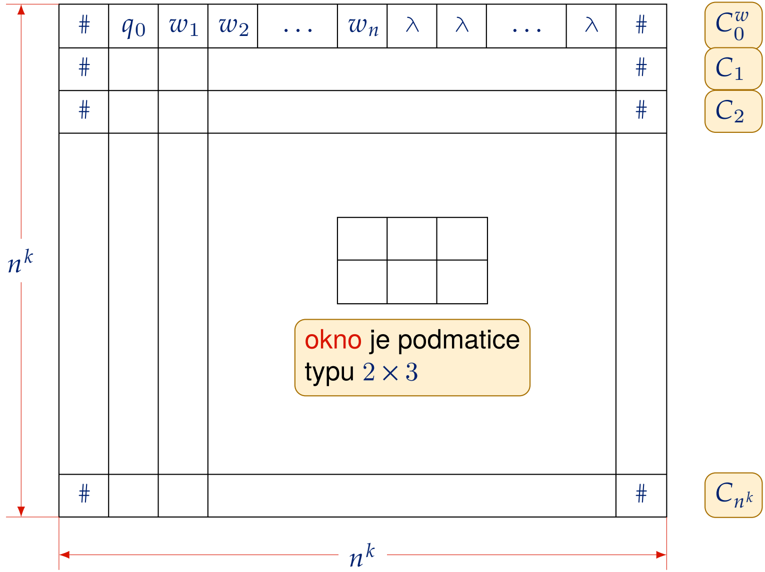
$$\begin{aligned}\varphi_{\text{start}} = {} & x_{1,1,\#} \wedge x_{1,2,q_0} \\ & \wedge x_{1,3,w_1} \wedge x_{1,4,w_2} \wedge \cdots \wedge x_{1,n+2,w_n} \\ & \wedge x_{1,n+3,\lambda} \wedge \cdots \wedge x_{1,n^k-1,\lambda} \\ & \wedge x_{1,n^k,\#}\end{aligned}$$

Sémantika: Jeden z řádků obsahuje přijímající konfiguraci

- Předpokládejme, že M má jediný přijímající stav q_1
- Požadujeme, aby nějaká buňka v T obsahovala q_1

$$\varphi_{\text{accept}} = \bigvee_{1 \le i,j \le n^k} x_{i,j,q_1}$$

Okno



q_0 w_1 w_2 ... w_n λ λ ... λ
C_0^w
C_1
C_2
C_{n^k}
n^k
okno je podmatice typu 2×3

Přípustné okno

okno podmatice T typu 2×3

(i,j)-okno má v levém horním rohu buňku $T[i,j]$

přípustné okno je takové, které se může vyskytnout jako část přechodu z jedné konfigurace do další přechodovou funkcí δ

Uvažme následující přechodovou funkci

$$\delta(q_2, \mathtt{a}) = \{(q_4, \mathtt{c}, L), (q_3, \mathtt{b}, R)\}$$
$$\delta(q_4, \mathtt{b}) = \{(q_2, \mathtt{a}, L)\}$$

podle ní je následující přechod přípustný

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

q_2	a	c
b	c	c

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

q_2	a	c
b	c	c

a	c	λ
c	c	λ

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

q_2	a	c
b	c	c

a	c	λ
c	c	λ

c	λ	λ
c	λ	λ

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

q_2	a	c
b	c	c

a	c	λ
c	c	λ

c	λ	λ
c	λ	λ

λ	λ	λ
λ	λ	λ

Přípustná okna

#	λ	λ	b	q_2	a	c	λ	λ	λ	#
#	λ	λ	q_4	b	c	c	λ	λ	λ	#

Uvedený přechod ukazuje přípustnost následujících oken

#	λ	λ
#	λ	λ

λ	λ	b
λ	λ	q_4

λ	b	q_2
λ	q_4	b

b	q_2	a
q_4	b	c

q_2	a	c
b	c	c

a	c	λ
c	c	λ

c	λ	λ
c	λ	λ

λ	λ	λ
λ	λ	λ

λ	λ	#
λ	λ	#

Další přípustná okna

- Uvažme následující přechodovou funkci

$$\delta(q_2, \mathtt{a}) = \{(q_4, \mathtt{c}, L), (q_3, \mathtt{b}, R)\}$$
$$\delta(q_4, \mathtt{b}) = \{(q_2, \mathtt{a}, L)\}$$

- Podle ní jsou též následující okna přípustná

#	q_2	a
#	b	q_3

λ	λ	q_4
λ	q_2	λ

λ	b	c
λ	b	q_4

a	b	a
q_3	b	a

b	c	c
b	c	c

a	q_4	b
q_2	a	a

Nepřípustná okna

- Uvažme následující přechodovou funkci

$$\delta(q_2, \mathtt{a}) = \{(q_4, \mathtt{c}, L), (q_3, \mathtt{b}, R)\}$$
$$\delta(q_4, \mathtt{b}) = \{(q_2, \mathtt{a}, L)\}$$

- Podle ní následující okna přípustná nejsou

#	q_2	a
#	q_2	a

q_3	λ	q_4
λ	q_2	λ

λ	b	c
λ	c	b

b	a	a
q_2	b	a

q_3	c	c
b	c	c

a	a	q_4
q_2	a	a

Hlavní vlastnost přípustných oken

Lemma

Předpokládejme, že

- *první řada tableau obsahuje počáteční konfiguraci se vstupem w a*
- *všechna okna v tableau jsou přípustná.*

Pak každá řádka tableau je konfigurací, jež následuje předchozí konfiguraci dle přechodové funkce.

Důkaz

- Indukcí dle pořadí řádku
- Řádka 1 je konfigurací z předpokladu
- Předpokládejme dvě následující řady i a $i+1$
- Předpokládejme, že řádek i je konfigurací
- Ukážeme, že i řádek $i+1$ je konfigurací

Přípustná okna tableau (důkaz)

Ověříme všechny symboly a konfigurace na řádku i

- $a = \#$
  - Přípustné okno okopíruje $\#$ z horní řady do dolní
  - Symboly $\#$ se vyskytují na okrajích každého řádku
- $a \in \Sigma$ v buňce, která nesousedí se symbolem stavu
  - a je uprostřed horní řady okna, jež neobsahuje symbol stavu
  - Z přípustnosti okna plyne, že a je též uprostřed spodní řady okna
  - a je v témž sloupci i v řadě $i + 1$
- $a = q \in Q$
  - q je uprostřed horní řady nějakého okna
  - Z přípustnosti okna plyne, že stav i okolní symboly jsou upraveny dle přechodové funkce δ

Je-li na řádce i konfigurace, pak na řádce $i + 1$ je konfigurace, která následuje i-tou konfiguraci dle přechodové funkce.

Množina přípustných oken

- Úprava dle přechodové funkce je lokální
  - Změna se týká jen 5 oken
- Zbylá okna jen kopírují horní řadu do spodní
  - má tedy spodní řadu shodnou s horní
- Množinu W přípustných oken lze zkonstruovat se znalostí přechodové funkce δ
  - Pro dané okno je možné zkontrolovat, je-li přípustné
  - $|W| \leq |S|^6$ což je konstanta, je-li M pevně daný

Zakódování přípustných oken

- Následující formule reprezentuje fakt, že (i, j)-okno je přípustné

$$\mathrm{legal}_{i,j} = \bigvee_{\substack{s_1,\dots,s_6 \\ \text{je přípustné okno}}} \big(x_{i,j,s_1} \wedge x_{i,j+1,s_2} \wedge x_{i,j+2,s_3} \wedge x_{i+1,j,s_4} \wedge x_{i+1,j+1,s_5} \wedge x_{i+1,j+2,s_6}\big)$$

- $\mathrm{legal}_{i,j}$ má konstantní velikost
- $\mathrm{legal}_{i,j}$ má ekvivalentní KNF konstantní velikosti
  - Lze zkonstruovat s použitím distributivity $\vee$ a $\wedge$

Sémantika: Každý řádek následuje předchozí dle přechodové funkce

$$\varphi_{\text{move}} = \bigwedge_{1 \leq i,j \leq n^k} \text{legal}_{i,j}$$

- Použijeme-li KNF ekvivalentní formuli $\text{legal}_{i,j}$, pak φ_{move} je KNF

Velikost φ

- Počet proměnných je $(n^k)^2 \cdot |S|$
- $|S|$ je konstantní, tedy počet proměnných je $O(n^{2k})$
- Konstrukci φ lze provést v polynomiálním čase

Velikost φ je polynomiální v n.

- Z konstrukce plyne, že

φ je splnitelná, právě když $w \in A$.

3-SAT

SAT (připomenutí)

Splnitelnost (SAT)	
Instance:	Formule φ v KNF.
Otázka:	Je formule φ splnitelná?

Věta

SAT je NP*-úplný problém.*

3-SAT

3-KNF formule φ je v 3-KNF, pokud je v KNF a každá klauzule obsahuje právě 3 literály

3-SAT	
Instance:	Formule φ v 3-KNF.
Otázka:	Je φ splnitelná?

Věta

Problém 3-SAT je NP-úplný.

NP-úplnost 3-SATu

3-SAT patří do třídy NP

- Týž polynomiální verifikátor jako pro SAT
- Ověřuje, jestli je daná formule φ splněna daným ohodnocením $\mathbf{a}$

3-SAT je NP-těžký

- SAT je polynomiálně převoditelný na 3-SAT

KNF φ —— převedeme ——→ 3-KNF ψ

φ je splnitelná $\Longleftrightarrow$ ψ je splnitelná

Převod SAT na 3-SAT

- Mějme KNF φ, jež má
  - n proměnných $x_1, \ldots, x_n$
  - m klauzulí $C_1, \ldots, C_m$
- Popíšeme konstrukci 3-KNF ψ, pro kterou platí

$$\varphi \text{ je splnitelná} \iff \psi \text{ je splnitelná}$$

- Pro každou klauzuli C_j, $j = 1, \ldots, m$
  - Podle potřeby přidáme nové proměnné
  - Sestrojíme konjunkci nových klauzulí α_j
  - Ohodnocení splňující C_j může být rozšířeno na model α_j
  - Ohodnocení splňující α_j splňuje C_j
- Definujeme $\psi = \bigwedge_{j=1}^{m} \alpha_j$
- Rozlišíme několik případů dle velikosti klauzule C_j

Prázdná klauzule

$C_j = \perp$ je prázdná klauzule

- C_j není splnitelná $\implies$ φ je nesplnitelná
- α_j je konjunkcí všech klauzulí délky 3 na nových proměnných y_1, y_2 a y_3

$$\begin{aligned}\alpha_j = {} & (y_1 \vee y_2 \vee y_3) \wedge (y_1 \vee y_2 \vee \neg y_3) \\ & \wedge (y_1 \vee \neg y_2 \vee y_3) \wedge (y_1 \vee \neg y_2 \vee \neg y_3) \\ & \wedge (\neg y_1 \vee y_2 \vee y_3) \wedge (\neg y_1 \vee y_2 \vee \neg y_3) \\ & \wedge (\neg y_1 \vee \neg y_2 \vee y_3) \wedge (\neg y_1 \vee \neg y_2 \vee \neg y_3)\end{aligned}$$

C_j ani α_j nemají model.

Klauzule délky 1

$C_j = l$ pro nějaký literál l

- Přidáme dvě nové proměnné y_1 a y_2

$$\begin{aligned}\alpha_j = {} & (l \vee y_1 \vee y_2) \wedge (l \vee y_1 \vee \neg y_2) \\ & \wedge (l \vee \neg y_1 \vee y_2) \wedge (l \vee \neg y_1 \vee \neg y_2)\end{aligned}$$

Nechť $\mathbf{a}$ je ohodnocení, které přiřazuje hodnotu l, y_1 a y_2.

$\mathbf{a}$ splňuje $C_j \iff \mathbf{a}$ splňuje α_j

Klauzule délky 2

$C_j = l_1 \vee l_2$ pro nějaké literály l_1 a l_2

- Přidáme novou proměnnou y

$$\alpha_j = (l_1 \vee l_2 \vee y) \wedge (l_1 \vee l_2 \vee \neg y)$$

Nechť $\mathbf{a}$ je ohodnocení, které přiřazuje hodnotu l_1, l_2 a y.

$$\mathbf{a} \text{ splňuje } C_j \iff \mathbf{a} \text{ splňuje } \alpha_j$$

Klauzule délky 3

$C_j = l_1 \vee l_2 \vee l_3$ pro nějaké literály l_1, l_2 a l_3

- Ponecháme C_j beze změny

$$\alpha_j = C_j$$

Nechť $\mathbf{a}$ je ohodnocení, které přiřazuje hodnotu l_1, l_2 a l_3.

$$\mathbf{a} \text{ splňuje } C_j \iff \mathbf{a} \text{ splňuje } \alpha_j$$

Klauzule velikosti $k > 3$ (myšlenka)

$C_j = l_1 \vee \cdots \vee l_k$ pro $k > 3$ a nějaké literály $l_1, \ldots, l_k$

Myšlenka:

- Přidáme novou proměnnou y a položíme

$$\beta = (l_1 \vee l_2 \vee y) \wedge (\neg y \vee l_3 \vee \cdots \vee l_k)$$

- Ohodnocení $\mathbf{a}'$, které splňuje β, splňuje také C_j
- Pokud $\mathbf{a}$ splňuje C_j
  - Můžeme rozšířit $\mathbf{a}$ na ohodnocení $\mathbf{a}'$, které splňuje β
  - Stačí zvolit vhodnou hodnotu y
- Dělení opakujeme, dokud nemáme jen klauzule velikosti 3

Klauzule velikosti $k > 3$

$C_j = l_1 \vee \cdots \vee l_k$ pro $k > 3$ a nějaké literály $l_1, \ldots, l_k$

- Přidáme nové proměnné $y_1, \ldots, y_{k-3}$

$$\alpha_j = (l_1 \vee l_2 \vee y_1) \wedge (\neg y_1 \vee l_3 \vee y_2) \wedge \cdots \wedge (\neg y_{i-2} \vee l_i \vee y_{i-1}) \\ \wedge \cdots \wedge (\neg y_{k-4} \vee l_{k-2} \vee y_{k-3}) \wedge (\neg y_{k-3} \vee l_{k-1} \vee l_k)$$

Ukážeme, že

1. pokud ohodnocení $\mathbf{a}'$ splňuje α_j, pak splňuje i C_j
2. pokud $\mathbf{a}$ přiřazuje hodnoty literálům $l_1, \ldots, l_k$ a pokud $\mathbf{a}$ splňuje C_j, pak jej lze rozšířit na model α_j

Model α_j splňuje C_j

$$\alpha_j = (l_1 \vee l_2 \vee y_1) \wedge (\neg y_1 \vee l_3 \vee y_2) \wedge \cdots \wedge (\neg y_{i-2} \vee l_i \vee y_{i-1})$$
$$\wedge \cdots \wedge (\neg y_{k-4} \vee l_{k-2} \vee y_{k-3}) \wedge (\neg y_{k-3} \vee l_{k-1} \vee l_k)$$

- Uvažme situaci, kde všechny literály l_i mají hodnotu 0
- Obdržíme formuli

$$\alpha'_j = y_1 \wedge (\neg y_1 \vee y_2) \wedge \cdots \wedge (\neg y_{i-2} \vee y_{i-1}) \wedge \cdots \wedge (\neg y_{k-4} \vee y_{k-3}) \wedge \neg y_{k-3}$$

- Dostáváme, že $\alpha'_j \models y_1$, tedy

$$\alpha'_j \equiv y_1 \wedge y_2 \wedge (\neg y_2 \vee y_3) \wedge \cdots \wedge (\neg y_{i-2} \vee y_{i-1}) \wedge \cdots \wedge (\neg y_{k-4} \vee y_{k-3}) \wedge \neg y_{k-3}$$

Model α_j splňuje C_j

$$\alpha'_j \equiv y_1 \wedge y_2 \wedge (\neg y_2 \vee y_3) \wedge \cdots \wedge (\neg y_{i-2} \vee y_{i-1}) \wedge \cdots \wedge (\neg y_{k-4} \vee y_{k-3}) \wedge \neg y_{k-3}$$

- Platí $\alpha'_j \models y_2$
- Indukcí odvodíme $\alpha'_j \models y_{k-3}$
- Navíc $\alpha'_j \models \neg y_{k-3}$
- Dohromady tedy $\alpha'_j \models \perp$
- Jinými slovy, α'_j je nesplnitelná
- Každý model α_j musí splnit nějaký z literálů $l_1, \ldots, l_k$

Každý model α_j splňuje C_j

Modely C_j lze rozšířit na modely α_j

- Nechť $\mathbf{a}$ je model C_j
  - $\mathbf{a}$ splňuje některý z literálů $l_1, \ldots, l_k$
  - Označme p index některého splněného literálu
  - Tedy $\mathbf{a}(l_p) = 1$
- Popíšeme model $\mathbf{a}'$ formule α_j, který rozšiřuje $\mathbf{a}$
  - $\mathbf{a}'(x_i) = \mathbf{a}(x_i)$, $i = 1, \ldots, n$
  - Určíme navíc hodnoty proměnných $y_1, \ldots, y_{k-3}$

Rozšíření modelu C_j

- Pro $i = 1, \ldots, k-3$, položíme $\mathbf{a}'(y_i) = \begin{cases} 1 & i \leq p-2 \\ 0 & i > p-2 \end{cases}$
- α_j je potom splněná ohodnocením $\mathbf{a}'$

$$\begin{aligned} \alpha_j = (l_1 \vee l_2 \vee \boxed{y_1}) \wedge (\neg y_1 \vee l_3 \vee \boxed{y_2}) \wedge \cdots \\ \wedge (\neg y_{p-3} \vee l_{p-1} \vee \boxed{y_{p-2}}) \wedge (\neg y_{p-2} \vee \boxed{l_p} \vee y_{p-1}) \\ \wedge (\boxed{\neg y_{p-1}} \vee l_{p+1} \vee y_p) \wedge \cdots \\ \wedge (\boxed{\neg y_{k-4}} \vee l_{k-2} \vee y_{k-3}) \wedge (\boxed{\neg y_{k-3}} \vee l_{k-1} \vee l_k) \end{aligned}$$

Každý model C_j lze rozšířit na model α_j vhodným ohodnocením proměnných $y_1, \ldots, y_{k-3}$

Vlastnosti konstrukce

- ψ lze sestrojit v polynomiálním čase pro danou KNF φ
- Je-li $\mathbf{a}$ modelem φ
  - $\mathbf{a}$ splňuje všechny klauzule C_j
  - Pro každé j lze $\mathbf{a}$ rozšířit na model α_j
  - Rozšíření jsou vzájemně nezávislá
  - Dohromady dostáváme, že $\mathbf{a}$ lze rozšířit na model ψ
- Je-li $\mathbf{a}'$ model ψ
  - $\mathbf{a}'$ splňuje všechny podformule α_j
  - Z toho plyne, že $\mathbf{a}'$ splňuje všechny klauzule C_j
  - $\mathbf{a}'$ je tedy modelem φ

φ je splnitelná $\iff$ ψ je splnitelná.