Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCA

Štefan Bigoš

Hornovské formule

Katedra teoretické informatiky a matematické logiky

Vedúci diplomovej práce: *doc. RNDr. Ondřej Čepek, Ph.D.*
Študijný program: *Informatika, Teoretická informatika*

Chcel by som poďakovať vedúcemu práce doc. RNDr. Ondřejovi Čepkovi, Ph.D. za jeho ochotu, čas, podklady a cenné pripomienky. Ďalej by som rád poďakoval rodičom za nepochopiteľne bezhraničnú podporu, Vrbovi za pomocnú ruku, ostatnej rodine a priateľom za spoluúčasť.

Vyhlasujem, že som svoju diplomovú prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím so zapožičiavaním práce.

V Prahe dňa 10. decembra 2009 Štefan Bigoš

Obsah

Názov práce: *Hornovské formule*
Autor: *Štefan Bigoš*
Katedra: *Katedra teoretické informatiky a matematické logiky*
Vedúci diplomovej práce: *doc. RNDr. Ondřej Čepek, Ph.D.*
e-mail vedúceho: *Ondrej.Cepek@mff.cuni.cz*

Abstrakt: *Jednou z aktuálne najštudovanejších podmnožín Booleovských funkcií sú Hornovské funkcie. V nich je mnoho nezodpovedaných otázok v probléme ich minimalizácie (nájdenie najkompaktnejšej ekvivalentnej reprezentácie).*
Tak ako Kronus [11] rozšíril poznatky v oblasti Hornovských formulí o neobvyklú mieru počtu zdrojových množín na základe teórie z relačných databáz, tak táto práca posúva jeho snahu o krok ďalej a tentokrát na základe teórie z hypergrafov zadefinuje ďalšie tri bežne nepoužívané miery a odvodí ich súvislosti a vlastnosti.
Ďalším sledovaným problémom je zložitosť Hornovskej minimalizácie pri obmedzení počtu literálov v terme. Doteraz najsilnejším tvrdením bol dôkaz o tom, že NP-ťažkosť sa zachováva aj keď obmedzíme tento počet na tri. Okrem nájdenia nezrovnalosti v dôkaze a jeho podrobnému rozobraniu, zadefinujem aj problém ktorý je ekvivalentný tomu, ktorý sa nachádza v článku. Tým dám priestor na prípadnú opravu dôkazu v budúcnosti.
Najväčším prínosom by potom malo byť vyplnenie vzniknutej medzery dokázaním o čosi slabšieho tvrdenia. Totiž, že problém Hornovskej minimalizácie zostáva NP-ťažký aj pri obmedzení počtu literálov v terme na 4.
Kľúčové slová: *Booleovské funkcie, Hornovská minimalizácia, Miera formule, Počet literálov v terme*

Title: *Horn Formulas*
Author: *Štefan Bigoš*
Department: *Department of Theoretical Computer Science and Mathematical Logic*
Supervisor: *doc. RNDr. Ondřej Čepek, Ph.D.*
Supervisor's e-mail address: *Ondrej.Cepek@mff.cuni.cz*

Abstract: *One of the recently most studied subsets of Boolean functions are Horn functions. There is quite a lot of open problems which concern their minimization (finding the smallest equivalent representation).*
As Kronus [11] has expanded knowledge in the area of Horn minimization by unusual measure (number of source sets) based on the theory of relational databases, in a similiar way this work is trying to expand this knowledge by another three unusual measures (and their properties, relations) based on the theory of hypergraphs.
Another recently studied problem is Horn minimization with a constraint on the number of literals in one term. Up to now the best result was a proof that Horn minimization remains intractable even when this number is restricted to be at most three. After pointing out a mistake in this proof, a new problem of finding HV-cover will be presented, which is equivalent to the studied one. Alltogether this could be the base for future correction of the original proof.
Then the biggest contribution of this work will be filling in the just created gap in the theory by a weaker proposition, that Horn minimization problem remains intractable even with a restriction on the number of literals per term to be at most four.
Keywords: *Boolean functions, Horn Minimization, Formula measure, Number of literals per term*

Kapitola 1

Úvod

Booleovská funkcia je svojou definíciou veľmi jednoduchý abstraktný matematický pojem. V praxi je však teória Booleovských funkcií veľmi dobre aplikovateľná na veľmi široku škálu (i nanajvýš aktuálnych) problémov. Zrejme je to spôsobené práve jej jednoduchosťou čo do definície, jej dostatočnou mierou abstrakcie. V odbore zložitosti bol napríklad prvý problém, pre ktorý bola dokázaná NP-úplnosť, problém splniteľnosti Booleovskej funkcie zadanej v tvare CNF.
A práve to viedlo k tomu, že sa medzi značne všeobecnými Booleovskými funkciami hľadali podtriedy, ktoré by mohli mať lepšie vlastnosti (napríklad kvadratické, kubické, acyklické, kvazi-acyklické, CQ-funkcie, Hornovské, skryte Hornovské, ...). V tejto práci som sa zameral práve na Hornovské funkcie. Tie zaujímajú medzi Booleovskými funkciami tak trochu výnimočné postavenie. A to práve z dôvodu, že odstraňujú obrovskú nevýhodu obecných skrývajúcu sa v zložitosti problému splniteľnosti a zároveň sú dostatočne všeobecné na to, aby v nich boli formalizované mnohé praktické problémy, obzvlášť z oboru umelej inteligencie a relačných databáz.
Pri narábaní s Hornovskými funkciami je kľúčovou otázkou voľba reprezentácie. V otázke *typu* reprezentácie sa na úkor menej typických (obvody, pravdivostná tabuľka) bude práca zaoberať len s funkciami vyjadrenými v najbežnejšej forme - vo forme formulí a to v *disjunktívnej normálnej forme* (DNF). Táto disjunktívna normálna forma však nie je jednoznačne určená, existuje typicky exponenciálne mnoho ekvivalentných vyjadrení danej funkcie. Pričom či už z pohľadu priestorovej, ale i časovej zložitosti je jasné, že práca s kompaktnejšou reprezentáciou je výhodnejšia. Preto jedným z najaktuálnejších problémov v tejto oblasti je problém nazývaný Horn Minimization (HM) - minimalizácie reprezentácie Hornovských formulí, na ktorý sa práca zameriava. Pri riešení tohto problému existuje viacero prístupov.
Jednou z možností je aproximácia. Mimo iných autorov sa týmto zaoberal aj Bhattacharya a kol. [2], kde okrem aproximácie skúmali pri probléme minimalizácie aj málo skúmané postupy, ako napríklad pridávanie nových premenných za účelom celkového zmenšenia reprezentácie i vzhľadom k počtu pôvodných premenných.
Alternatívnym prístupom je hľadanie podtried Hornovských formulí, pre ktoré by bol problém minimalizácie riešiteľný v polynomiálnom čase. Najznámejším prípadom sú kvadratické funkcie (algoritmus na minimalizáciu uvádza napríklad aj Kronus [11]). Tie sú však už značne obmedzené a v praxi málo využiteľné. Preto je snaha nájsť všeobecnejšie vyjadrenie. Tak napríklad Hammer a Kogan [9] uviedli už značne použiteľnejšiu množinu *acyklických* funkcií a ešte aj jej nadmnožinu *quasi-acyklických* funkcií. Jeden z najnovších výsledkov v tejto oblasti priniesol Boros a kol. [3] uvedením takzvaných *CQ-funkcií.*

Ďalšou intenzívne študovanou podtriedou problému Hornovskej minimalizácie je trieda Hornovských formulí s obmedzením počtu literálov na jeden term. Takéto obmedzenia však typicky neprinášajú úspech. To sa vie napríklad u vyššie spomínaného problému splniteľnosti. Pri probléme Hornovskej minimalizácie nie je tento aspekt preskúmaný úplne do detailov. Značne totiž záleží na tom, čo považujeme za parameter, ktorý chceme minimalizovať. A to je práve oblasť, ktorá je náplňou tejto práce.
V prvej kapitole zhrniem základné definície týkajúce sa Hornovských funkcií, ako aj vety a nástroje, ktoré potom budem využívať vo všetkých ostatných kapitolách.
Ako bolo spomenuté vyššie, problém minimalizácie je značne závislý od miery, ktorou meriam jej dĺžku. A práve druhá kapitola si vzala za cieľ preskúmať čo najširšie spektrum rôznych mier minimality, dokonca aj takých ktoré boli doteraz uvažované len mimo svet Hornovských funkcií. Pozrie sa na ich vlastnosti, vzájomné závislosti a rozoberie problém minimalizácie pre každú z nich.
Ďalšie kapitoly sa venujú problému minimalizácie pri obmedzení počtu literálov v terme. V prípade, kedy používame ako mieru počet literálov v celej formuli, ukázali Čepek, Kučera [6], že aj pri obmedzení počtu literálov v terme na 5, zostáva tento problém NP-ťažký. Ak mierou minimalizácie je počet termov vo formuli, ukázali Boros, Čepek [4] že sa dá počet literálov na term obmedziť až na číslo 3. Práve rozbor tohoto dôkazu bude náplňou tretej kapitoly. Ako sa ukáže, dôkaz obsahuje chybu. Uvediem protipríklad, dôkladne preskúmam silu dokázaných tvrdení. Pokúsim sa množstvo užitočných pomocných tvrdení uvedených v tomto dôkaze použiť na zkonštruovanie ekvivalentného problému hľadania tzv. HV-pokrytia. Okrem demonštrácie komplikovanosti situácie tak dám priestor a prostriedky na prípadné budúce dokázanie toho, čo chcel pôvodný dôkaz dokázať.
Nie je známy žiaden iný výsledok, ktorý by ukázal NP-ťažkosť minimalizácie v počte termov pri obmedzení na akýkoľvek vyšší počet literálov v terme. Štvrtá kapitola si zaumieni túto práve vzniknutú medzeru vyplniť a dokázať aspoň slabšie tvrdenie, v ktorom sa neobmedzí počet literálov v terme na 3 (ako to činil pôvodný dôkaz), ale na 4. Ukáže to redukciou z NP-ťažkého problému známeho ako Set Covering. Respektíve z jeho modifikácie, ktorá má obmedzenie mohutnosti množín na 3, nazvaného 3-Set Covering. Dôkaz NP-ťažkosti tohoto problému je taktiež súčasťou tejto kapitoly. Rovnako aj dôkazy NP-ťažkosti ďalších problémov, ktoré sú nevyhnutné pre tento dôkaz.

1.1 Základné pojmy

V tejto sekcii objasním základné pojmy z oblasti Booleovských a Hornovských funkcií. Uvediem aj niektoré základné vlastnosti popisovaných objektov vo forme Lemmat. Aby som zabránil neúmernemu navyšovaniu rozsahu práce, nebudem uvádzať dôkazy všetkých tvrdení. Obzvlášť ak je dôkaz veľkého rozsahu a dá sa nájsť v inej literatúre. V takomto prípade sa typicky odkážem na zdroj, kde sa dá nájsť a uvediem aspoň náznak dôkazu. Podobný postup budem voliť aj v nasledujúcich kapitolách.
Pri tvorbe tejto kapitoly som vychádzal takmer výlučne z [5].

1.1.1 Booleovské funkcie

Základnou definíciou je

Definícia 1.1 ***Booleovská funkcia*** *je zobrazenie*

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

Malými písmenami z konca abecedy budem označovať logické premenné, typicky x, y. Môžu teda nadobúdať hodnoty $0, 1$, ktoré sú ekvivalentami logických hodnôt *false*, *true*. Pri operáciach s logickými premennými budem používať na označenie disjunkcie $x \vee y$, konjukcie $x \wedge y$, negácie $\bar{x}$. Výskyt premennej alebo jej negácie budem nazývať **literál**, pričom výskyt premennej je *pozitívny literál*, výskyt jej negácie *negatívny literál.*

Definícia 1.2 ***Booleovská formula****:*

a) každá atomická formula (literál) je booleovská formula,

b) ak A, B sú booleovské formule, potom aj $\bar{A}, A \vee B, A \wedge B$ *sú booleovské formule*

Slovom **term** budem označovať konjukciu literálov, v ktorej sa žiadna premenná nevyskytne zároveň v pozitívnom i negatívnom literále. Na reprezentáciu Booleovských funkcií budem používať práve formule, a to takmer výlučne v tvare:
Disjunktívna normálna forma (DNF) je disjunkcia termov.

Dovoliť si to môžem preto, lebo platí nasledujúce tvrdenie:

Veta 1.3 *Každá Booleovská funkcia sa dá vyjadriť v tvare DNF.*

Dôkaz. Majme funkciu $f(n)$ na n premenných $x_1, \ldots x_n$. Definujme tabuľku:

x_1	x_2	$\ldots$	x_n	$f(n)$
0	0	$\cdots$	0	0
0	0	$\cdots$	1	1
$\vdots$				$\vdots$
1	1	$\cdots$	0	0
1	1	$\cdots$	1	1

Kde v jednotlivých stĺpcoch máme hodnoty premenných, v poslednom stĺpci hodnotu funkcie. Výsledná DNF bude disjunkciou termov, kde každý term odpovedá jednému riadku tabuľky, kde má funkcia hodnotu *jedna*. Pri tvorbe termu z jedného riadku sa použije postup, kde sa hodnota *nula* nahradí negatívnou premennou a hodnota *jedna* pozitívnou. Takže v prípade hodnôt, ktoré sú naznačené v tabuľke by výsledná DNF vyzerala

$$(\bar{x_1} \wedge \bar{x_2} \wedge \ldots \wedge x_n) \vee \ldots \vee (x_1 \wedge x_2 \wedge \ldots \wedge x_n)$$

□

Poznámka: Z tabuľky je pekne vidieť, že počet rôznych Booleovských funkcií nad n premennými je $2^{(2^n)}$. Riadky totiž odpovedajú rôznym ohodnoteniam premenných, tých je 2^n. Booleovská funkcia potom môže mať na každom riadku buď hodnotu *jedna* alebo *nula*. Takýchto rôznych kombinácií je práve $2^{(2^n)}$.

Zavediem si ešte jedno značenie. Formule v tvare DNF budem označovať kaligrafickým písmom veľkými písmenami, typicky $\mathcal{F}$, $\mathcal{G}$, $\mathcal{H}$. Symbolom $\hat{\mathcal{F}}$ budem označovať množinu termov DNF $\mathcal{F}$.

Pre funkcie f a g budem písať $f \leq g$, ak $f = 1 \Rightarrow g = 1$. Presnejšie povedané, ak pre všetky hodnoty premenných platí $\forall (x_1, ..., x_n) \in \{0,1\}^n : f(x_1, x_2, \dots, x_n) = 1 \Rightarrow g(x_1, x_2, \dots, x_n) = 1$, kde f a g sú definované na n premenných. Každý term taktiež odpovedá Booleovskej funkcii, takže toto značenie sa vzťahuje aj na termy. Dokonca v prípade, kedy bude pre nejaký term T a funkciu f platiť, že $T \leq f$, takýto term označím pojmom **implikant** funkcie f. Term T sa nazýva **primárny implikant** funkcie f, ak je implikantom, a po vypustení ľubovoľného literálu prestáva byť implikantom (T sa nedá skrátiť). Ak je T implikantom f a nie je primárny, potom existuje T' tak, že $T \leq T' \leq f$, T' je primárny implikant f. V takomto prípade hovoríme, že T' **absorbuje** T.
Poznámka: V DNF je každý term implikantom reprezentovanej funkcie.

Ak je však každý term v DNF *primárnym* implikantom reprezentovanej funkcie, potom túto DNF nazvem **primárnou**. DNF $\mathcal{F}$ reprezentujúcu f nazvem **irredundantnou**, ak vypustením ľubovoľného termu z $\mathcal{F}$ dostanem DNF, ktorá už nereprezentuje f.
Hovoríme, že $\mathcal{F}$ je **kanonická DNF**, ak obsahuje všetky primárne implikanty reprezentovanej funkcie.

Termy T_1 a T_2 majú **konflikt** v premennej x, ak $x \in T_1$ a $\bar{x} \in T_2$ (alebo naopak). Povieme, že T_1 a T_2 majú **konsenzus**, ak majú konflikt v práve jednej premennej. Potom ak $T_1 = Ax, T_2 = B\bar{x}$, tak $cons(T_1, T_2) = A \wedge B$.
Ako hovorí Veta 1.4, ktorú uvediem nižšie, na získanie kanonickej DNF slúži

Konsenzuálna metóda: Na vstupe má DNF $\mathcal{F}$. Kým sa $\mathcal{F}$ môže meniť, opakuj:

1) vykonaj všetky absorbcie, ktoré sa vykonať dajú
2) najdi ľubovoľné dva termy majúce konsenzus, a ak nie je absorbovaný termom, ktorý už je v $\mathcal{F}$ prítomný, tak ho pridaj do $\mathcal{F}$

Veta 1.4 (Quinn) *Nech $\mathcal{F}$ je DNF reprezentujúca f. Výstupom konsenzuálnej metódy so vstupom $\mathcal{F}$ je kanonická DNF reprezentujúca f.*

Dôkaz. Sporom. Nech $\mathcal{G} = R_1 \vee R_2 \vee \ldots \vee R_k$ je výstupom konsenzuálnej metódy a nech P je primárny implikant f, pričom $P \neq R_i, \forall i$. Definujme $\mathcal{P} = \{\text{term } Q | Q \leq P, \forall i : Q \nleq R_i\}$. Keďže $P \in \mathcal{P}$, tak $\mathcal{P} \neq \emptyset$. Nech je teda P^* najdlhší term v $\mathcal{P}$.

- $|P^*| = n$ (všetky premenné)
  $P^* = 1 \Rightarrow P = 1 \Rightarrow f = 1$
  $P^* = 1 \Rightarrow \forall i : R_i = 0 \Rightarrow f = 0$, Spor

- $|P^*| \leq n - 1$
  Vieme teda, že $\exists x : x \notin P^*$. Termy $xP^*, \bar{x}P^*$ nie sú v $\mathcal{P}$, pretože P^* bol najdlhší. Keďže $P^* \leq P$, potom aj $xP^* \leq P$ a $\bar{x}P^* \leq P$. Potom ale musia tieto dva termy porušovať vlastnosť $\forall i : Q \nleq R_i$. Musia teda existovať termy R_i, R_j také, že platí $xP^* \leq R_i, R_i = Ax, A \subseteq P^*$ a zároveň $\bar{x}P^* \leq R_j, R_j = B\bar{x}, B \subseteq P^*$. Potom ale tieto termy majú nutne konsenzus $cons(R_i, R_j) = A \wedge B$, kde $A \wedge B \subseteq P^*$. Keďže je to konsenzus prvkov z $\mathcal{G}$, musel byť metódou pridaný, alebo absorbovaný nejakým $R_k \in \mathcal{G}$. Potom ale $R_k \subseteq A \wedge B \subseteq P^*$ a teda $P^* \leq R_k$ a to je spor s náležením do $\mathcal{P}$. $\square$

1.1.2 Hornovské formule

Term T sa nazýva **Hornovský**, ak obsahuje nanajvýš jeden negatívny literál.
DNF $\mathcal{F}$ sa nazýva **Hornovská**, ak sa skladá len z Hornovských termov.
Booleovská funkcia f sa nazýva **Hornovská**, ak má aspoň jednu reprezentáciu pomocou Hornovskej DNF. Term T sa nazýva **čiste Hornovský**, ak obsahuje práve jeden negatívny literál. Analogicky definujeme čiste Hornovskú DNF, čiste Hornovskú funkciu.

V prípade Hornovských funkcií je dobre známym faktom (napríklad [7]), že každý primárny implikant (čiste) Hornovskej funkcie je (čiste) Hornovský. Takže každá primárna DNF (čiste) Hronovskej funkcie je (čiste) Hornovská.

Pre zjednodušenie značenia pri práci s termami si zavediem zjednodušené značenie, kde namiesto konjunkcie premenných $x \wedge y \wedge \bar{z}$ budem písať len jednoducho $xy\bar{z}$. Ak budem mať množinu premenných $S = \{x_1, x_2, ..., x_n\}$ a premennú z, budem namiesto $\bigwedge_{x_i \in S} x_i \wedge \bar{z}$ písať len jednoducho $S\bar{z}$. Na niektorých miestach sa v prípade takýchto termov budem na množinu S pozitívnych literálov odkazovať ako na *ľavú stranu* termu a na premennú z, teda na premennú ktorá vystupuje ako negatívna, ako na *hlavu termu*.

V úvode som spomínal, že Hornovské formule majú práve tú výhodu, že pre nich je algoritmus na zistenie splniteľnosti polynomiálny. Problém splniteľnosti je však definovaný na konjuktívnej normálnej forme (CNF). Čitateľa nemusí zaujímať čo CNF je, podstané je totiž to, že v prípade DNF (a v celej práci sa zaoberám práve výlučne DNF) je tento problém ekvivalentný problému falzifikovateľnosti, označovaný **FALS**, ktorý hľadá ohodnotenie, v ktorom má funkcia hodnotu 0. (Čiže v podstate zisťuje, či daná formula je tautológiou.)
Pre názornosť uvediem aspoň naivnú verziu algoritmu na FALS DNF:

1) Kým existuje záporný lineárny term $\bar{x}_i$:
   (a) dosaď $x_i = 1$
   (b) z formule vyhoď termy obsahujúce $\bar{x}_i$
   (c) z termov obsahujúcich x_i tento literál vyhoď
   Ak som týmto vytvoril prázdny term, potom KONIEC - formula nie je falzifikovateľná (je tautológiou)
2) Zvyšné premenné ohodnoť nulou.

Tento algoritmus pracuje v čase $O(nl)$, kde n je počet premenných, l je počet literálov. Existuje aj o málo chytrejší algoritmus, ktorý pracuje v čase $O(l)$.

Veľmi dôležitým a často používaným pojmom pri štúdiu Hornovských funkcií je **forward chaining**. Je to procedúra, pomocou ktorej sa dá overiť, či daný term je implikantom zadanej čiste Hornovskej funkcie. Na vstupe má čiste Hornovskú DNF $\mathcal{F}$ reprezentujúcu funkciu f, podmnožinu Q jej premenných a na výstupe dá množinu R, ktorá je nadmnožinou množiny Q. Vágne povedané, množina R reprezentuje množinu všetkých premenných, ktoré sú "odvoditeľné" z premenných, ktoré sú v Q. Pričom táto "odvoditeľnosť" sa dá nahliadnuť napríklad tak, že pri pokuse falzifikovať $\mathcal{F}$ je R množinou všetkých premenných, ktorých ohodnotenie na 1 je vynútené ohodnotením premenných z Q na 1.

Forward chaining. Na vstupe má čiste Hornovskú DNF $\mathcal{F} = \bigvee_{i=1}^{m} S_i \bar{x_{j_i}}$ a podmnožinu premenných Q. Potom:

1) $R := Q$

2) Kým existuje i tak, že $S_i \subseteq R$ a $x_{j_i} \notin R$, potom $R := R \cup \{x_{j_i}\}$

Výstupom procedúry forward chaining je množina R. Všeobecne v ďalšom texte výsledok procedúry forward chaining nad DNF $\mathcal{F}$ s počiatočnou množinou Q budem označovať symbolom $\mathbf{R}_{\mathcal{F}}(\mathbf{Q})$.

O užitočnosti tejto procedúry, hovorí následujúce Lemma. Keďže ho budem v ďalšom texte hojne používať, nebudem sa na neho zakaždým odkazovať. Jeho dôkaz neuvediem, dá sa nájsť v mnohých publikáciách, napríklad v [11], [8].

Lemma 1.5 *Čiste Hornovský term $Q\bar{y}$ je implikantom čiste Hornovskej funkcie f zadanej čiste Hornovskou DNF $\mathcal{F}$ práve vtedy, ak $y \in R_{\mathcal{F}}(Q)$.*

Na základe tejto vety si zadefinujem ešte dva pojmy. Premenná x je v $\mathcal{F}$ **odvoditeľná** z množiny S, ak $x \in R_{\mathcal{F}}(S)$. Term $S\bar{x}$ je **odvoditeľný** z $\mathcal{F}$, ak $x \in R_{\mathcal{F}}(S)$.

Ďalšia z užitočných vlastností Hornovských DNF je vyjadrená v nasledujúcom Lemma.

Lemma 1.6 *Pre každú Hornovskú DNF sa dá v čase $O(l^2)$ nájsť ekvivalentná primárna irredundantná DNF.*

Dôkaz. Majme Hornovskú DNF $\mathcal{F}$. Ukážem, ako k nej nájsť primárnu irredundantnú DNF $\mathcal{F}''$, ktorá bude ekvivalentná.
Dôležitým faktom je, že pre každú Hornovskú DNF $\mathcal{H}$ a term $S\bar{x}$ vieme v lineárnom čase $O(l)$ (l je počet literálov) zistiť, či $S\bar{x} \leq \mathcal{H}$. A to tak, že v $\mathcal{H}$ položíme všetky premenné z S ako rovné 1, premennú x položíme rovnú 0 - túto novú DNF nazvyme $\mathcal{H}'$. Tým zabezpečíme, že $S\bar{x}$ je pravda. Ak má byť splnená podmienka $S\bar{x} \leq \mathcal{H}$, musí byť nutne aj $\mathcal{H}'$ pravda, a to pre každé ohodnotenie. Inými slovami, nesmie byť falzifikovateľná. A to zistíme tak, že pustíme lineárny algoritmus FALS na $\mathcal{H}'$. A ak má byť $S\bar{x} \leq \mathcal{H}$, musí algoritmus skončiť s odpoveďou NO (teda s odpoveďou "vstupná DNF nie je falzifikovateľná").
Takže teraz si zoberieme $\mathcal{F}$, z nej vytvoríme primárnu $\mathcal{F}''$. A to tak, že z každého termu T spravím primárny nasledovne. Skúsim z neho odstrániť nejaký literál a zistím, či aj po jeho odstránení platí $T \leq \mathcal{F}$ (horeuvedeným lineárnym algoritmom). Ak áno, odstránim tento literál. Takto postupne vyskúšam všetky literály. Táto časť spustí pre každý literál horeuvedený algoritmus, to nám dáva zložitosť tejto časti $O(l^2)$.
Teraz z primárnej $\mathcal{F}'$ spravím primárnu irredundantú $\mathcal{F}''$. A to tak, že pre každý term zistím, či nie je redundantný. Najprv $\mathcal{F}'' := \mathcal{F}'$. Potom pre každý term T polož $\mathcal{F}'' := \mathcal{F}'' \setminus T$ a skús či $T \leq \mathcal{F}''$. Ak áno, tak term je nadbytočný, nechaj ho mimo $\mathcal{F}''$. Ak nie, tak term T vráť do $\mathcal{F}''$. Pre každý term som pustil horeuvedený $O(l)$ algoritmus, takže ak t je počet termov, tak táto časť beží v čase $O(lt)$.
□

Kapitola 2

Hornovská Minimalizácia (HM) a netradičné miery

V tejto kapitole zadefinujem problém Hornovskej minimalizácie (HM). Veľmi vágne sa dá povedať, ze je to snaha pre zadanú DNF $\mathcal{F}$ nájsť jej minimálnu reprezentáciu, to jest DNF $\mathcal{F}'$ ekvivalentú $\mathcal{F}$ takú, aby jej dĺžka bola najmenšia možná. Otázkou je, čo je to "dĺžka formule".
A práve na túto otázku sa v tejto kapitole pokúsim čo najkomplexnejšie odpovedať. Skúsim nájsť všetky odpovede, ktoré pre Hornovské DNF dávajú nejaký dobrý zmysel.
Vo vedeckej literatúre sa pod pojmom *Horn Minimization* väčšinou myslí jej minimalizácia vzhľadom k najprirodzenejšej miere - počtu termov vo formuli. Veľmi často používanou je ešte aj celkový počet literálov vo formuli. Obe tieto miery sú celkom bežne používané aj na meranie dĺžky všeobecných formulí v tvare DNF (nie nutne Hornovských).
Špeciálny tvar Hornovskej DNF však umožňuje dívať sa na veľkosť jej DNF aj na základe iných kritérií, ktoré u ostatných nemajú obdobu. Takto napríklad Kronus [11] zadefinoval mieru, ktorú vo svojej práci označil ako $|\mathcal{F}|_r$. Tam aj na základe článku z oblasti relačných databáz, ktorý publikoval Maier [12], vytvoril dôkaz o polynomiálnej zložitosti minimalizácie Hornovských formulí vzhľadom k tejto miere. Podobný prístup som zvolil aj ja v tejto kapitole a na základe článku z oblasti hypergrafov, ktorý publikoval Ausiello a kol. [1], som zadefinoval ďalšie tri miery, ktoré dávajú aj vo svete Hornovských funkcií dobrý zmysel.
Potom s pomocou viet a lemmat dokázaných v [11] a [1] odvodil aj ich vlastnosti a vzájomné vzťahy.

Pod pojmom **miera** čiste Hornovskej formule budeme rozumieť nejaké zobrazenie z množiny všetkých Hornovských DNF do množiny prirodzených čísel. **Dĺžka** (veľkosť) formule v miere $|\mathcal{F}|_x$ bude teda hodnota tejto miery pre túto DNF. Povieme, že formula $\mathcal{F}$ je **minimálna** vzhľadom k miere $|\mathcal{F}|_x$, ak neexistuje žiadna iná DNF $\mathcal{F}'$ tak, aby platilo $|\mathcal{F}'|_x < |\mathcal{F}|_x$.

Teraz môžem prejsť k definícii, ktorá je v tejto práci kľúčová.

Definícia 2.1 *Problémom **Hornovskej minimalizácie (HM)** v miere $|\mathcal{F}|_x$ rozumieme:*

Zadanie *Hornovská DNF $\mathcal{F}$, prirodzené číslo k, miera $|\mathcal{F}|_x$.*

Otázka *Existuje DNF ekvivalentná s $\mathcal{F}$ tak, aby na nej bola definovaná miera $|\mathcal{F}|_x$ a jej dĺžka bola nanajvýš k?*

Niekedy budem označovať problém Hornovskej minimalizácie vzhľadom k miere $|\mathcal{F}|_x$ ako HM-x.
Vzhľadom k podmienkam, ktoré som stanovil pre miery, môžem dokázať nasledujúce tvrdenie.

Lemma 2.2 *Problém Hornovskej minimalizácie vzhľadom k ľubovoľnej miere $|\mathcal{F}|_x$ je v NP.*

Dôkaz. Majme inštanciu problému HM-x, ktorá mala na vstupe Hornovskú DNF $\mathcal{F}$ a mieru $|\mathcal{F}|_x$, a na ktorú bola kladná odpoveď a je daná (nie nutne Hornovská) DNF $\mathcal{F}'$ ako "certifikát".
Najprv (v polynomiálnom čase) zistíme hodnotu $|\mathcal{F}'|_x$. Ak je väčšia ako k, zamietneme $\mathcal{F}'$ ako certifikát.
Pre každý term T' z $\mathcal{F}'$ skonštruujeme v polynomiálnom čase primárny implikant T'' DNF $\mathcal{F}$ tak, aby platilo $T' \leq T'' \leq \mathcal{F}$ (odstraňovaním literálov - viď dôkaz Lemma 1.6 - až na to, že novovznikajúci term budem testovať v tomto prípade nie voči $\mathcal{F}'$, ktorá nie je Hornovská, ale voči $\mathcal{F}$, ktorá je Hornovská). Nech $\mathcal{F}''$ je disjunkcia týchto primárnych implikantov. Z predchádzajúcej kapitoly vieme, že každá primárna DNF Hornovskej funkcie je Hornovská. Takže ak $\mathcal{F}''$ nie je Hornovská, zamietneme $\mathcal{F}'$ ako certifikát.
Teraz si overíme, či $\mathcal{F}$ a $\mathcal{F}''$ reprezentujú tú istú funkciu. Najprv overíme, či $\mathcal{F} \leq \mathcal{F}''$. A to tak, že pre každý term T zistíme, či $T \leq \mathcal{F}''$. Ako som ukázal v dôkaze Lemma 1.6, toto vieme v čase $O(l), l$ je počet literálov (keďže sú obe Hornovské). Prebehne to pre každý term, to nám dáva $O(lt), t$ je počet termov. Podobne overíme, či $\mathcal{F}'' \leq \mathcal{F}$.
Ak $\mathcal{F}''$ prešla všetkými testami, tak máme skutočne riešenie problému minimalizácie. Ak niektorý z testov zlyhal, potom môžeme $\mathcal{F}''$ ako certifikát odmietnuť.
□

Poznámka: Problém hľadania minimálnej DNF v prípade všeobecných formulí (nie Hornovských) je NP-ťažký pre každú mieru. Stačí si uvedomiť, že FALS je špeciálny prípad minimalizácie. Ak totiž formula nie je falzifikovateľná, je tautologicky rovná 1 ($x \vee \bar{x}$). Predpokladám samozrejme, že v ľubovoľnej miere je každá formula dlhšia ako tautologická jednička.

2.1 Pomocné tvrdenia

V tejto sekcii uvediem základné definície a tvrdenia týkajúce sa mier minimality, ktoré mi potom v sekciách nasledujúcich umožnia dokázať vety, ktoré sú výsledkom celej tejto kapitoly. Pre prehľadnosť tak učiním v súlade s formalizáciou z [11] a z [1].

Pre konjunkciu $D = \bigwedge_{x \in S} x$ označím symbolom $\mathbf{I(D)}$ množinu premenných tvoriacich jej literály, čiže v tomto prípade $I(D) = S$.

Definícia 2.3 *Pre Hornovskú DNF $\mathcal{F}$ reprezentujúcu Booleovskú funkciu f a pozitívne konjunkcie D a E poviem, že D **implikuje** E podľa $\mathcal{F}$, ak platí*

$$\forall x \in I(E) \setminus I(D) : (D\bar{x} \leq f).$$

*Toto budem značiť $D \to E$ podľa $\mathcal{F}$. Ak zároveň $E \to D$ podľa $\mathcal{F}$, potom je D **ekvivalentné** E podľa $\mathcal{F}$, čo budeme označovať $D \leftrightarrow E$. Pre ľubovoľnú konjunkciu S pozitívnych literálov označím symbolom $E_{\mathcal{F}}(S)$ množinu tých termov $\mathcal{F}$, ktorých ľavá strana je ekvivalentná S podľa $\mathcal{F}$. Symbolom $e_{\mathcal{F}}(S)$ označím množinu ľavých strán termov z $E_{\mathcal{F}}(S)$.*

Teraz popíšem procedúru nazvanú *modifikovaný forward chaining*. Jej priebeh nad $\mathcal{F}$ je úplne rovnaký ako priebeh normálneho forward chaining až na to, že pri behu nad vstupnou množinou S môže používať len tie termy z $\mathcal{F}$, ktorých ľavá strana nie je ekvivalentná s S. Inak povedané, používa len termy T, pre ktoré platí $T \notin E_{\mathcal{F}}(\bigwedge_{x \in S} \bar{x})$. Výslednú množinu modifikovaného forward chaining nad DNF $\mathcal{F}$ so vstupnou množinou S budem označovať $R'_{\mathcal{F}}(S)$.

Definícia 2.4 *Nech $\mathcal{F}$ je čiste Hornovská DNF reprezentujúca f, C a D sú pozitívne konjunkcie také, že $C \to D$ podľa $\mathcal{F}$. Poviem, že C* ***priamo implikuje*** *D podľa $\mathcal{F}$ (s označením $C \xrightarrow{*} D$), ak existuje irredundantná čiste Hornovská DNF $\mathcal{G}$ ekvivalentná s $\mathcal{F}$, pre ktorú platí $I(D) \subseteq R'_{\mathcal{G}}(I(C))$.*

Definícia 2.5 *Majme čiste Hornovskú DNF $\mathcal{F}$ reprezentujúcu f. Zdrojovú množinu S nazvem* ***nadbytočnou****, ak $\exists T : S \leftrightarrow T, S \xrightarrow{*} T$.*
$\mathcal{F}$ nazvem ***LR-minimálnou****, ak je primárna irredundantná a zároveň neobsahuje žiadne nadbytočné zdrojové množiny.*

Lemma 2.6 *Nech $\mathcal{F}$ je čiste Hornovská primárna iredundantná DNF a S je jej zdrojová množina. Nech T je pozitívna konjunkcia taká, že $S \leftrightarrow T$ a zároveň $S \xrightarrow{*} T$ (podľa $\mathcal{F}$). Potom DNF $\mathcal{G}$ definovaná následovne*

$$\textit{Prvkom } \mathcal{G} \textit{ je} \begin{cases} U\bar{x} & \textit{ak } U\bar{x} \in \mathcal{F}, U \neq S \\ T\bar{x} & \textit{ak } S\bar{x} \in \mathcal{F} \end{cases}$$

je ekvivalentná $\mathcal{F}$.

Dôkaz. Kompletný dôkaz tohoto tvrdenia je v [11]. Na tomto mieste uvediem aspoň myšlienku dôkazu.
Vieme, že $S \xrightarrow{*} T$, takže pri odvodzovaní T z S nemusí byť použitý ani jediný term, ktorý má ako ľavú stranu množinu S. Takže ak z $\mathcal{F}$ odstránim všetky termy, čo majú ako ľavú stranu S, term T bude v tejto novovzniknutej DNF aj naďalej odvoditeľný. Takže ak v $\mathcal{G}$ pustím forward chaining nad S, množina T bude vo výsledku zahrnutá. A ak pustím v $\mathcal{G}$ forward chaining nad T, zahrnú sa mi tam určite všetky premenné, ktoré by sa mi v $\mathcal{F}$ zahrnuli nad S (keďže som nahradil všetky výskyty S za T). A keďže jediný rozdiel medzi $\mathcal{F}$ a $\mathcal{G}$ sú len termy čo majú ako ľavú stranu S alebo T, dokázal som tým ich ekvivalenciu. (Najprv $\mathcal{F} \leq \mathcal{G}$, potom aj $\mathcal{G} \leq \mathcal{F}$.)
□

V ďalšom texte budeme budeme pod pojmom *eliminácia nadbytočnej zdrojovej množiny* rozumieť práve operáciu popísanú v znení Vety 2.6. Poznamenám, že touto operáciou sa nemení počet termov. Inak povedané - výsledná DNF $\mathcal{G}$ má rovnaký počet termov ako $\mathcal{F}$. Počet zdrojových množín sa však znížil o jedničku (nahradili sme všetky výskyty S, nahradili sme ich zdrojovou množinou T, ktorá bola obsiahnutá už i v $\mathcal{F}$).

2.2 Miery minimality

V tejto kapitole sa teda konečne dostávam k tomu, že pre danú čiste Hornovskú DNF $\mathcal{F}$ zadefinujem miery, podľa ktorých sa môže posudzovať jej veľkosť. Hlavným účelom tejto definície bude snaha ukázať, aký ťažký bude problém Hornovskej minimalizácie vzhľadom k týmto mieram. Takže zároveň s definíciou miery zadefinujem aj pojmy, ktoré charakterizujú vlastnosť "byť minimálny vzhľadom k tejto miere". Ak nebude možné aby došlo k omylu, budem tieto pojmy v ďalšom texte navzájom zamieňať (napríklad sa budem odkazovať na mieru SAM, namiesto $|\mathcal{F}|_a$).

Definícia 2.7 *Majme čiste Hornovskú primárnu irredundantú DNF $\mathcal{F}$ reprezentujúcu Hornovskú funkciu f o n premenných. Povieme, že $\mathcal{F}$ je:*

- *Source-minimum (**SM**) - ak je $\mathcal{F}$ minimálna vzhľadom k počtu rôznych zdrojových množín, tj. vzhľadom k miere $|\mathcal{F}|_s$ definovanej nasledovne*[1]*:*

$$|\mathcal{F}|_s = |\{S | S\bar{x} \in \hat{\mathcal{F}}\}|.$$

- *Term-minimum (**TM**) - ak je $\mathcal{F}$ minimálna vzhľadom k počtu termov, tj. vzhľadom k miere*

$$|\mathcal{F}|_t = |\hat{\mathcal{F}}|$$

- *Source-term-minimum (**STM**) - ak je minimálna vzhľadom k miere*

$$|\mathcal{F}|_{st} = |\mathcal{F}|_s + |\mathcal{F}|_t$$

- *Source-area-minimum (**SAM**) - ak je minimálna vzhľadom k veľkosti zdrojovej oblasti, tj. k miere*

$$|\mathcal{F}|_a = \sum_{T | \exists x : T\bar{x} \in \hat{\mathcal{F}}} |T|$$

- *Optimum (**O**) - ak je minimálna vzhľadom k jej skrátenej reprezentácii, tj. k miere*

$$|\mathcal{F}|_o = |\mathcal{F}|_a + |\mathcal{F}|_t$$

- *Literal-minimum (**LM**) - ak je minimálna vzhľadom k celkovému počtu literálov, tj. k miere*

$$|\mathcal{F}|_l = \sum_{T \in \hat{\mathcal{F}}} |T|$$

Pri definícii som použil niekoľko nových výrazov, ktorých význam sa dal dovtípiť, ale pre istotu ich zadefinujem presne. Konjunkciu pozitívnych literálov T nazvem **zdrojová množina** DNF $\mathcal{F}$, ak $\exists x : T\bar{x} \in \hat{\mathcal{F}}$, kde x je premenná $\mathcal{F}$. Termínom **zdrojová oblasť** DNF $\mathcal{F}$ budem mať na mysli súčet mohutností všetkých jej zdrojových množín (teda ak sa jedna zdrojová množina vyskytne vo viacerých termoch, do zdrojovej oblasti ju započítam len raz).

Pre jednoduchosť si zavediem ešte nasledujúce značenie. Pre DNF $\mathcal{F}$ označím písmenom a veľkosť jej zdrojovej oblasti (teda a $(= a_{\mathcal{F}}) = |\mathcal{F}|_a$), písmenom m počet termov (teda m $(= m_{\mathcal{F}}) = |\mathcal{F}|_t$), písmenom s počet rôznych zdrojových množín (teda s $(= s_{\mathcal{F}}) = |\mathcal{F}|_s$). Písmenom n budem v súlade so zvyškom práce označovať počet premenných.

Pozastavím sa ešte nad tým, prečo som dal do definície podmienku primárnosti a irredundantnosti. Platí nasledujúce Lemma:

[1] Kronus v [11] označuje túto mieru symbolom $|\mathcal{F}|_r$

Lemma 2.8 *Majme Hornovskú DNF $\mathcal{F}$, mieru $|\mathcal{F}|_x$ ľubovoľnú z vyššie uvedených (tj. $|\mathcal{F}|_x \in \{|\mathcal{F}|_s, |\mathcal{F}|_t, |\mathcal{F}|_{st}, |\mathcal{F}|_a, |\mathcal{F}|_o, |\mathcal{F}|_l\}$). Medzi všetkými DNF, ktoré sú ekvivalentné $\mathcal{F}$ a zároveň minimálne vzhľadom k miere $|\mathcal{F}|_x$ môžeme nájsť v kvadratickom čase Hornovskú DNF $\mathcal{F}'$, ktorá je primárna a irredundantná.*

Dôkaz. Majme predpoklady tvrdenia, potom vieme nájsť DNF (nie nutne Hornovskú), ktorá je ekvivalentná $\mathcal{F}$ a minimálna vzhľadom k $|\mathcal{F}|_x$. Z nej vieme na základe dôkazu Lemma 1.6 (a s rovnakou poznámkou u tohto použitia ako v prípade dôkazu Lemma 2.2) skonštruovať v čase $O(l^2)$ ekvivalentnú primárnu irredundantú DNF $\mathcal{F}'$. Keďže je primárna, je aj Hornovská. Tvorili sme ju tak, že sme postupne odstraňovali termy a literály.
Rozborom prípadov u jednotlivých mier sa dá veľmi jednoducho rozmyslieť, že odstránením termu alebo literálu sa nezvyšuje hodnota tejto miery.
□

Takže teraz viem, že medzi minimálnymi DNF viem vždy nájsť primárnu irredundnantnú. Preto sa v definícii SM (SAM, TM, STM, O, LM) môžem obmedziť len na tie z nich, ktoré túto podmienku spĺňajú.

Príklad 2.9 *Majme nasledujúce Hornovské DNF. (Ich rozdelenie zátvorkami na dve časti je len z dôvodu prehľadnosti.)*

a) *Primárna irredundantná DNF $\mathcal{F}$.*

$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee cd\bar{e}) \vee (e\bar{f} \vee f\bar{e} \vee e\bar{g} \vee e\bar{h} \vee gh\bar{e} \vee ghk\bar{l})$$

b) *Ekvivalentná SM formula,*

$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee ab\bar{e}) \vee (e\bar{f} \vee f\bar{e} \vee e\bar{g} \vee e\bar{h} \vee gh\bar{e} \vee ghk\bar{l})$$

ktorá vznikne z $\mathcal{F}$ nahradením klauzule $cd\bar{e}$ klauzulou $ab\bar{e}$.

c) *Ekvivalentná TM formula,*

$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee cd\bar{e}) \vee (e\bar{f} \vee f\bar{g} \vee f\bar{h} \vee gh\bar{e} \vee ghk\bar{l})$$

ktorá vznikne z $\mathcal{F}$ nahradením klauzulí $f\bar{e}$, $e\bar{g}$, $e\bar{h}$ klauzulami $f\bar{g}$, $f\bar{h}$.

d) *Ekvivalentná STM formula,*

$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee ab\bar{e}) \vee (e\bar{f} \vee f\bar{g} \vee f\bar{h} \vee gh\bar{e} \vee ghk\bar{l})$$

ktorá vznikne z $\mathcal{F}$ kombináciou oboch vyššie uvedených nahradení.

e) *Ekvivalentná SAM formula,*

$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee ab\bar{e}) \vee (e\bar{f} \vee f\bar{e} \vee e\bar{g} \vee e\bar{h} \vee gh\bar{e} \vee fk\bar{l})$$

ktorá vznikne z SM formule z bodu b) nahradením klauzule $hgk\bar{l}$ klauzulou $fk\bar{l}$.

f) *Ekvivalentná O formula,*

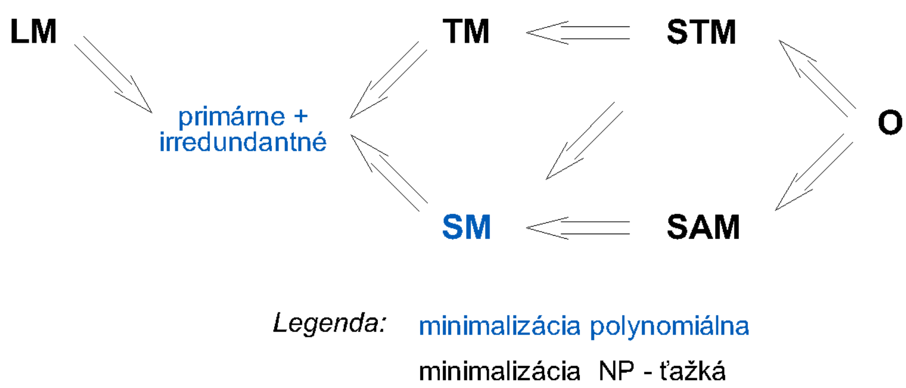
$$(ab\bar{c} \vee ab\bar{d} \vee c\bar{a} \vee d\bar{b} \vee ab\bar{e}) \vee (e\bar{f} \vee f\bar{g} \vee f\bar{h} \vee gh\bar{e} \vee fk\bar{l})$$

ktorá vznikne z STM formule z bodu d) nahradením klauzule $hgk\bar{l}$ klauzulou $fk\bar{l}$.

Poznámka: To, že sú formule z príkladu ekvivalentné sa dá overiť procedúrou forward chaining, ich minimalita vzhľadom k danej miere "hrubou silou".

2.3 Vzájomná závislosť

V tejto časti budem skúmať vzťahy medzi jednotlivými mierami. Ako sa ukáže, tieto miery sú veľmi silno navzájom prepojené. Výsledkom tejto kapitoly potom bude dôkaz, že platia všetky vzťahy medzi mierami ako sú uvedené na Obrázku 2.1 (kde napríklad šípka $STM \Rightarrow TM$ znamená, že ak je $\mathcal{F}$ STM, tak je zárověň aj TM). Z obrázku je napríklad vidieť, že pre každú funkciu f sa dá nájsť DNF $\mathcal{F}$, ktorá ju reprezentuje a je minimálna vzhľadom ku všetkým uvedeným mieram okrem LM.



Obr. 2.1: Vzťahy medzi mierami

Na dôkaz týchto závislosti však potrebujem ešte niekoľko pomocných tvrdení.

Lemma 2.10 *Majme čiste Hornovskú DNF $\mathcal{F}$. $\mathcal{F}$ je Source minimum práve vtedy keď $\mathcal{F}$ je LR-minimálna.*

Dôkaz.
$\Longrightarrow$ Táto implikácia je priamym dôsledkom Lemma 2.6. Ak by totiž nebola LR-minimálna, mohli by sme eliminovaním nadbytočnej zdrojovej množiny znížiť $|\mathcal{F}|_s$ o jedničku. To by bol spor s predpokladom, že $\mathcal{F}$ je SM.
$\Longleftarrow$ Kompletný dôkaz je uvedený v [11]. Konkrétne tam je ukázané, že ak máme DNF $\mathcal{G}$, ktorá je SM a irredundantnú DNF $\mathcal{F}$, ktorá je s $\mathcal{G}$ ekvivalentná a existuje taká konjunkcia pozitívnych literálov že $|E_{\mathcal{G}}(S)|_s < |E_{\mathcal{F}}(S)|_s$, potom existujú $U \in e_{\mathcal{F}}(S), V \in e_{\mathcal{F}}(S), U \neq V$ také, že $U \xrightarrow{*} V$ podľa $\mathcal{F}$.
A z tohto už dôkaz priamo vyplýva. Ak by totiž $\mathcal{F}$ bolo LR-minimálne, ale nebolo SM, potom by existovala SM DNF $\mathcal{G}$ a nutne by teda pre nejaké S platilo, že $|E_{\mathcal{G}}(S)|_s < |E_{\mathcal{F}}(S)|_s$. Z toho by ale vyplývala existencia U a V takých, že $U \xrightarrow{*} V$ v $\mathcal{F}$, čo by bol spor s LR-minimalitou.
□

Lemma 2.11 *Majme ekvivalentné čiste Hornovské DNF $\mathcal{F}$ a $\mathcal{G}$, ktoré sú obe SM. Potom pre ľubovoľnú konjunkciu pozitívnych literálov S platí $|E_{\mathcal{F}}(S)| = |E_{\mathcal{G}}(S)|$. Naviac pre každé $X_i \in e_{\mathcal{F}}(S)$ existuje práve jedno $Y_j \in e_{\mathcal{G}}(S)$ také, že $X_i \xrightarrow{*} Y_j$ a zároveň $Y_j \xrightarrow{*} X_i$.*

Dôkaz. Kompletný dôkaz je opäť v [11]. Uvediem len náčrt myšlienky.
Ak by platilo, že pre nejaké S je $|E_{\mathcal{F}}(S)|_s < |E_{\mathcal{G}}(S)|_s$, potom by museli existovať dve rôzne

$Y_j, Y_k \in \mathcal{G}$ a také $X_i \in \mathcal{F}$, že $Y_j \xrightarrow{*} X_i$ a zároveň $Y_k \xrightarrow{*} X_i$. Ale pre X_i existuje také Y_l, že $X_i \xrightarrow{*} Y_l$. Mali sme $j \neq k$, takže aspoň jedno z j, k je rôzne od l, BÚNO $k \neq l$. Keďže za určitých podmienok (ktoré sú teraz splnené) platí tranzitivita $\xrightarrow{*}$, potom z predchádzajúceho vyplýva, že $Y_k \xrightarrow{*} Y_l$, čo by bol spor s tým, že $\mathcal{G}$ je SM.
□

Výsledkom tejto kapitoly je nasledujúca veta zhrňujúca implikácie medzi jednotlivými konceptami minimalizácie.

Veta 2.12 *Majme čiste Hornovskú DNF $\mathcal{F}$.*

a) Ak $\mathcal{F}$ je STM, potom $\mathcal{F}$ je SM.

b) Ak $\mathcal{F}$ je STM, potom $\mathcal{F}$ je TM.

c) Ak $\mathcal{F}$ je SAM, potom $\mathcal{F}$ je SM.

d) Ak $\mathcal{F}$ je O, potom $\mathcal{F}$ je SAM.

e) Ak $\mathcal{F}$ je O, potom $\mathcal{F}$ je STM.

Dôkaz.

a) (STM $\Rightarrow$ SM). Budem postupovať sporom. Nech $\mathcal{F}$ je teda STM, ale nie je SM. Podľa Lemma 2.10 nie je LR-minimálna a keďže je STM, tak je aj primárna irredundantná. Existuje teda aspoň jedna nadbytočná zdrojová množina S. Eliminovaním tejto zdrojovej množiny dostaneme pomocou Vety 2.6 DNF $\mathcal{G}$, ktorá je ekvivalentná $\mathcal{F}$. Dostali sme DNF, ktorá má rovnaký počet termov ako $\mathcal{F}$, ale počet zdrojových množín má o jedna menší než $\mathcal{F}$. To je spor s predpokladom, že $\mathcal{F}$ je STM.

b) (STM $\Rightarrow$ TM). Postupujme sporom. Predpokladajme teda, že $\mathcal{F}$ je STM ale nie je TM. $\mathcal{F}$ nie je TM, existuje teda TM DNF $\mathcal{G}$, ktorá má ostro menej termov. Podľa časti a) tejto vety je $\mathcal{F}$ SM, takže $\mathcal{G}$ má viac rovno zdrojových množín než $\mathcal{F}$. Keby však nastala rovnosť a $\mathcal{G}$ malo rovnaký počet zdrojových množín ako $\mathcal{F}$, tak vzhľadom k faktu, že $\mathcal{G}$ má ostro menej termov by muselo mať aj ostro menší súčet $s + m$ a to je spor s predpokladom, že $\mathcal{F}$ je STM.
Takže $\mathcal{G}$ má ostro viac zdrojových množín než $\mathcal{F}$ a teda nie je SM a podľa Lemma 2.10 nie je ani LR-minimálny. Keďže $\mathcal{G}$ je TM, je primárny a iredundantný, má teda nevyhnutne aspoň jednu nadbytočnú zdrojovú množinu. Majme DNF $\mathcal{H}$, ktoré dostaneme z $\mathcal{G}$ odstránením všetkých nadbytočných zdrojových množín. Podľa definície a Vety 2.6, $\mathcal{H}$ je LR-minimálne a ekvivalentné $\mathcal{G}$ (teda reprezentujúce tú istú funkciu). Zároveň je podľa Lemma 2.10 SM. Keďže podľa časti a) tejto vety aj $\mathcal{F}$ je SM, majú $\mathcal{H}$ a $\mathcal{F}$ rovnaký počet zdrojových množín. Zároveň $\mathcal{H}$ má ostro menej termov než $\mathcal{F}$ (pretože $\mathcal{G}$ má ostro menej termov než $\mathcal{F}$ a $\mathcal{H}$ sme dostali z $\mathcal{G}$ odstraňovaním nadbytočných zdrojových množín, ktoré nezvyšuje (dokonca zachováva) počet termov). $\mathcal{H}$ je ekvivalentné $\mathcal{F}$ a má menší súčet $|\mathcal{F}|_s + |\mathcal{F}|_t$ a to je spor s predpokladom, že $\mathcal{F}$ je STM.

c) (SAM $\Rightarrow$ SM). $\mathcal{F}$ je primárny irredundantný (keďže je SAM). Navyše $\mathcal{F}$ nemá žiadnu nadbytočnú zdrojovú množinu (inak by sme totiž odstránením tejto nadbytočnej zdrojovej množiny na základe Vety 2.6 dostali ekvivalentnú DNF, ktorá by mala menšiu zdrojovú oblasť než $\mathcal{F}$, čo by bol spor s predpokladom, že $\mathcal{F}$ je SAM). Takže $\mathcal{F}$ je LR-minimálny a teda podľa Lemma 2.10 je aj SM.

d) (O $\Rightarrow$ SAM). Postupujme sporom. Predpokladajme teda, že $\mathcal{F}$ je optimálna (minimálny súčet $a+m$), ale existuje ekvivalentná SAM DNF $\mathcal{G}$, ktorá má ostro menšiu zdrojovú oblasť než $\mathcal{F}$. Podľa bodu c) tejto vety je $\mathcal{G}$ SM. Zároveň aj $\mathcal{F}$ je SM (inak by nebola podľa Vety 2.10 LR-minimálna, existovala by teda nadbytočná zdrojová množina, jej odstránením by sme dostali DNF s menšou zdrojovou oblasťou a rovnakým počtom termov, čo by bol spor s predpokladom, že $\mathcal{F}$ je O).
Existuje teda bijekcia ϕ medzi zdrojovými množinami $\mathcal{F}$ a $\mathcal{G}$ taká, že $S \overset{*}{\rightarrow} \phi(S)$ a zároveň $\phi(S) \overset{*}{\rightarrow} S$ (Lemma 2.11). Nech $\mathcal{H}$ vznikne z $\mathcal{F}$ tak, že nahradíme všetky termy $S\bar{x}$ termami $\phi(S)\bar{x}$ (pre každú zdrojovú množinu S formule $\mathcal{F}$). Viacnásobným aplikovaním Vety 2.6 dostávame, že $\mathcal{H}$ je ekvivalentná $\mathcal{F}$. Zároveň má rovnaký počet termov ako $\mathcal{F}$, ale zdrojovú oblasť má rovnakú ako $\mathcal{G}$ (a tá je ostro menšia ako zdrojová oblasť $\mathcal{F}$). $\mathcal{H}$ má teda menší súčet $a+m$ než $\mathcal{F}$, a to je spor s predpokladom, že $\mathcal{F}$ je O.

e) (O $\Rightarrow$ STM). Budem postupovať podobne ako v dôkaze časti d) tejto vety. Takže opäť sporom - nech $\mathcal{F}$ je O, nie je STM, existuje STM DNF $\mathcal{G}$ ekvivalentné $\mathcal{F}$. Rovnako ako v dôkaze časti d) tejto vety dostaneme, že $\mathcal{F}$ je SM. Z dokázanej časti a) priamo dostávam, že aj $\mathcal{G}$ je SM. $\mathcal{F}$ nie je STM, $\mathcal{G}$ je STM, platí teda $s_{\mathcal{G}} + m_{\mathcal{G}} < s_{\mathcal{F}} + m_{\mathcal{F}}$. Obe DNF sú SM a teda $s_{\mathcal{G}} = s_{\mathcal{F}}$ a z toho plynie $m_{\mathcal{G}} < m_{\mathcal{F}}$. Keďže $\mathcal{F}$ je O, platí $a_{\mathcal{F}} + m_{\mathcal{F}} \leq a_{\mathcal{G}} + m_{\mathcal{G}}$. Z predchádzajúcich dvoch nerovností dostávame $a_{\mathcal{F}} < a_{\mathcal{G}}$. Obidve formule sú SM, máme teda podobne ako v d) bijekciu $\phi : S \overset{*}{\rightarrow} \phi(S), \phi(S) \overset{*}{\rightarrow} S$. Nech $\mathcal{H}$ vznikne tentokrát z $\mathcal{G}$ nahradením všetkých termov $S\bar{x}$ termami $\phi(S)\bar{x}$. Máme teda $m_{\mathcal{H}} = m_{\mathcal{G}}, a_{\mathcal{H}} = a_{\mathcal{F}}, a_{\mathcal{H}} + m_{\mathcal{H}} < a_{\mathcal{F}} + m_{\mathcal{F}}$, spor s predpokladom, že $\mathcal{F}$ je O.

□

Pozorovanie 2.13 *Ak nepočítame mieru $|\mathcal{F}|_l$, tak žiadne iné implikácie týkajúce sa vzťahov jednotlivých mier minimality neplatia (samozrejme okrem tých, ktoré získame tranzitivitou).*

Dôkaz. Podľa príkladu 2.9. Je vidieť, že

a) SM formula z bodu b) príkladu nie je TM (a teda ani STM),

b) TM formula z bodu c) príkladu nie je SM (a teda ani SAM, ani STM),

c) STM formula z bodu d) príkladu nie je SAM,

d) SAM formula z bodu e) príkladu nie je TM (a teda ani STM),

e) žiadna z formulí z bodov b) až e) príkladu nie je O.

□

Na obrázku 2.1 je okrem vzájomných vzťahov mier vyznačená aj zložitosť minimalizácie vzhľadom k príslušnej miere. Vďaka práve dokázanému tvrdeniu mi však na potvrdenie týchto zložitostí stačí ukázať nasledujúce.

Primárnosť a irredundantnosť sme schopní, ako som už uviedol vyššie, zabezpečiť v polynomiálnom čase.

Minimalizácia SM je uskutočniteľná v polynomiálnom čase. Vďaka Lemma 2.10 mi stačí eliminovať všetky nadbytočné zdrojové množiny, na čo mi v podstate stačí forward chaining.

Podrobný dôkaz polynomiálnej zložitosti tohoto problému včetne algoritmu sa dá nájsť v [11].

NP-úplnosť problému HM-l bola dokázaná vo viacerých publikáciách, napríklad [11], [6].

NP-úplnosť problémov HM-t a HM-a ukážem v kapitole 4. Tým budú zložitosti všetkých problémov odvodené.

Kapitola 3

3-HM: Chyba v existujúcom dôkaze

V probléme Hornovskej Minimalizácie sa, podobne ako v iných NP-úplných problémoch, skúmajú aj prípady, v ktorých sa oproti pôvodnému zadaniu urobí nejaké obmedzenie dúfajúc, že sa problém zjednoduší. Buď sa nájde polynomiálny algoritmus, ktorý takýto obmedzený problém rieši, alebo sa ukáže že aj pri obmedzeniach zostáva problém NP-úplný a teda nemá význam ani v budúcnosti hľadať polynomiálny algoritmus (aspoň do doby kým sa neukáže, že $P = NP$). Takéto obmedzenia sú typicky na mohutnosti množín, počty prvkov. Vo väčšine prípadov sa ukazuje ako kritický prechod z čísla 2 na číslo 3. Tak napríklad splniteľnoť obecných kvadratických DNF je jednoduchá, kým o probléme 3SAT sa už dávno vie, že je NP-úplný. U kvadratických je jednoduchý aj problém ich minimalizácie. Naproti tomu u Hornovských, napriek tomu že u nich máme splniteľnosť polynomiálnu, je problém minimalizácie NP-úplný. A to, ako som ukázal v predchádzajúcej kapitole, u takmer všetkých mier (teda s výnimkou miery počtu zdrojových množín). Tak sa hľadali určité obmedzenia, ktoré by u NP-ťažkých mier túto minimalizáciu zjednodušili. A u Hornovskej Minimalizácie sa ako najčastejší spôsob obmedzenia berie počet literálov na term. Pričom sa má za to, že čím nižšie číslo, tým väčšia šanca, že sa podarí nájsť polynomiálny aloritmus. Keďže Hornovské funkcie sú podmnožinou obecných je jasné, že pri obmedzení literálov na 2 vieme nájsť polynomiálny algoritmus. Potom je snaha nájsť čo najtesnejšiu hranicu v tomto obmedzení. V tomto úsilí sú výsledky rôzne v závislosti od miery, podľa ktorej minimalizujeme. Tak napríklad v prípade miery počtu literálov v celej DNF je najlepším doteraz dokázaným výsledkom, že aj pri obmedzení počtu literálov na term na 5 zostáva problém NP-úplný. Pre obmedzenia počtu literálov na 3 a 4 nevieme s istotou povedať ani či je to polynomiálne ani či NP-ťažké. V prípade miery, pri ktorej sa berie do úvahy počet termov, je situácia lepšia. Dokonca sa dosiahla úplne tesná hranica - dokázalo sa v [4] aj pre obmedzenie počtu na tri, že je tento problém NP-úplný. Avšak ako ukáže jeho rozbor, ktorý je obsahom tejto kapitoly, príslušný dôkaz obsahuje isté nezrovnalosti. V snahe túto nezrovnalosť odstrániť učiním jeho podrobný rozbor, ukážem širšie súvislosti dokázaných viet.

Dôkaz v snahe prevodu z Hamiltonovskej cesty v grafe G vytvorí definíciu jemu analogickej čiste Hornovskej funkcie f_G a jej DNF. Minimálna forma tejto Hornovskej DNF by potom mala generovať práve Hamiltonovskú cestu v pôvodnom grafe. Avšak, ako ukážem, množina podgrafov, ktorú táto f_G generuje, je o čosi širšia. V tejto práci som tieto podgrafy nazval HV-pokrytie. Vzhľadom k tomu, že je silný predpoklad, že problém Hornovskej minimalizácie je skutočne NP-ťažký aj pri tomto obmedzení, bolo mojou snahou problém ekvivalentného grafu funkcie f_G nadefinovať čo najpresnejšie, pretože sa dá očakávať, že aj problém generovania týchto mnou nadefinovaných HV-pokrytí bude taktiež NP-ťažký.

V prvej časti kapitoly uvediem pôvodný dôkaz NP-úplnosti HM pri obmedzení počtu literálov na term na tri, tak ako je uvedený v [4]. Potom ukážem, kde nastal v tomto dôkaze chybný predpoklad. Zvyšok dôkazu potom uvediem len schématicky, keďže vzhľadom k chybe už bude neplatný a nepoužiteľný.
V ďalšej časti potom rozoberiem, kde nastala chyba a ako to v skutočnosti je. Ukážem aj myšlienkový postup pri definovaní skutočných vlastností štruktúr uvedených v pôvodnom dôkaze.
V poslednej časti potom uvediem konečný výsledok - HV-pokrytie, včetne dôkazu, že problém hľadania minimálnej reprezentácie f_G je skutočne ekvivalentný hľadaniu nejakého HV-pokrytia pôvodného grafu G. Tento výsledok má potenciál použiteľnosti v tom, že ak sa ukáže, že je NP-úplný, tak na základe tejto kapitoly bude automaticky dokázaná aj NP-úplnosť Hornovskej minimalizácie pri obmedzení dĺžok termov na tri. Oproti 3-HM však môže mať výhodu v tom, že je to grafový problém a vzhľadom k značnej šírke vedomostí v oblasti grafových algoritmov je dosť možné, že sa jeho NP-ťažkosť dokáže ľahšie.

3.1 Existujúci dôkaz

V tejto časti uvediem prepis článku Boros, Čepek [4].
Ako som spomínal v úvode kapitoly, tento článok mal za hlavný cieľ ukázať NP-úplnosť problému Hornovskej minimalizácie aj pri obmedzení dĺžok termov na tri. Náležitosť do NP ukázana rovnako ako v Lemma 2.2. NP-ťažkosť ukázaná pomocou prevodu zo známeho NP-úplného problému hľadania (neorientovanej) Hamiltonovskej cesty v kubickom grafe.

Nech $G = (V, E)$ je ľubovoľný neorientovaný graf na n vrcholoch s m hranami. Každý vrchol grafu si označíme Booleovskou premennou x_i, takže $V = \{x_1, x_2, \ldots, x_n\}$. V celej kapitole budeme predpokladať, že graf G je jednoduchý, t.j. že nemá slučky (hrany (x, x)) ani viacnásobné hrany. Ku grafu G zadefinujeme k nemu pridruženú f_G pomocou nasledujúcej DNF

$$\mathcal{F}_G = (\bigvee_{x_i \in V} v\bar{x}_i) \vee (\bigvee_{(x_i, x_j) \in E} x_i x_j \bar{v}), \tag{3.1}$$

kde v označuje jednu dodatočnú Booleovskú premennú. Ako vidieť, DNF (3.1) pozostáva z $m + n$ termov a každý term pozostáva z maximálne 3 literálov. Každý term je zároveň očividne čiste Hornovský a primárny. Primárnosť sa dá nahliadnuť napríklad tak, že keď ľubovoľnú premennú nastavíme na hodnotu jedna, potom si ľahko vynútime hodnotu funkcie f_G nula tým, že aj všetkým ostatným premenným priradíme hodnotu jedna.
Graf $G = (V, E)$ označujeme ako *kubický*, ak každý jeho vrchol náleží presne trom hranám, tj. $\forall x \in V : \deg_G(x) = 3$.
Výsledkom celého článku potom malo byť dokázanie nasledujúceho tvrdenia:

Veta 3.1 *Nech $G = (V, E)$ je kubický neorientovaný graf. K nemu pridružená čiste Hornovská funkcia f_G má DNF reprezentáciu s maximálne $m + 2$ termami práve vtedy, ak v grafe G existuje Hamiltonovská cesta, tj. cesta prechádzajúca každým vrcholom práve raz.*

Najprv ukážeme, že každá DNF reprezentácia f_G musí mať minimálne $m + 2$ termov. Takže každá DNF, ktorá pozostáva z maximálne $m + 2$ termov musí byť nutne minimálna a mať teda presne $m + 2$ termov. Článok si dal za úkol aj polynomiálny prevod medzi

Hamiltonovskou cestou a minimálnou reprezentáciou funkcie f_G a naopak. To je práve časť, ktorá sa nakoniec nezdarila.
Nasledujúce Lemma charakterizuje všetky primárne implikanty funkcie f_G.

Lemma 3.2 *Nech $G = (V, E)$ je graf a uvažujme k nemu pridruženú funkciu f_G definovanú rovnicou* (3.1)*. Term T je primárnym implikantom funkcie f_G práve vtedy, keď je splnená jedna z nasledujúcich podmienok:*

a) $T = v\bar{x}_k$ pre nejaké $x_k \in V$, alebo

b) $T = x_i x_j \bar{x}_k$ pre nejaké $(x_i, x_j) \in E$ kde $k \notin \{i, j\}$, alebo

c) $T = x_i x_j \bar{v}$ pre nejaké $(x_i, x_j) \in E$.

Dôkaz. Podmnožinu $S \subseteq V$ vrcholov grafu G nazvem *stabilná*, ak žiadna hrana grafu G nemá oba svoje konce v tejto množine. Dá sa odpozorovať, že

- $R_{\mathcal{F}_G}(S) = S$, ak $S \subseteq V$ je stabilná,
- $R_{\mathcal{F}_G}(\{x_i, x_j\}) = V \cup \{v\}$, ak $(x_i, x_j) \in E$ a nakoniec
- $R_{\mathcal{F}_G}(\{v\}) = V \cup \{v\}$.

Z toho vyplýva že všetky implikanty kategórií a), b) a c) sú implikantami f_G a naviac vypustením ľubovoľného literálu vzniknú termy, ktoré už implikantami nebudú, takže sú to všetko skutočne primárne implikanty f_G.
Na druhej strane, ak je nejaký term T primárnym implikantom funkcie f_G, potom musí byť čiste Hornovský (keďže vieme, že každý primárny implikant čiste Hornovskej funkcie je Hornovský). Takže predpokladajme, že T je tvaru $T = Q\bar{y}$, kde Q je podmnožina premenných a $y \notin Q$. Potom $R_{\mathcal{F}_G}(Q) \neq Q$ (keďže musí obsahovať y), takže na základe vyššie uvedeného musí byť Q buď premenná v alebo nejaká hrana $(x_i, x_j) \in E$. Primalita T potom implikuje, že v prvom prípade $T = v\bar{y}$ (teda že T je kategórie a)) a v druhom prípade $T = x_i x_j \bar{y}$ (teda T je kategórie b) alebo c), v závislosti od y).
□

V súlade s vyššie zavedeným, pre hranu $e = (x_i, x_j) \in E$ grafu G, budem termy $x_i x_j \bar{v}$ a $x_i x_j \bar{x}_k$ písať skrátene $e\bar{v}$, $e\bar{x}_k$.
Každá primárna DNF $\mathcal{F}$ funkcie f_G musí na základe Lemma 3.2 pozostávať jedine z termov typu a), b) a c). Zavediem nasledujúce značenie.

- $S(\mathcal{F}) = \{x \in V | v\bar{x}$ je term v $\mathcal{F}\}$,
- $H_e(\mathcal{F}) = \{x \in V | e\bar{x}$ je term v $\mathcal{F}\}, e \in E$,
- $I(\mathcal{F}) = \{e \in E | H_e(\mathcal{F}) \neq \emptyset\}$,
- $T(\mathcal{F}) = \{e \in E | e\bar{v}$ je term v $\mathcal{F}\}$, a nakoniec
- $D(\mathcal{F}) = \bigcup_{e \in I(\mathcal{F})} H_e(\mathcal{F})$.

Takže použitím práve zavedeného značenia môže byť každá primárna DNF funkcie f_G zapísaná ako

$$\mathcal{F} = (\bigvee_{x \in S(\mathcal{F})} v\bar{x}) \vee (\bigvee_{e \in I(\mathcal{F})} \bigvee_{x \in H_e(\mathcal{F})} e\bar{x}) \vee (\bigvee_{e \in T(\mathcal{F})} e\bar{v}) \tag{3.2}$$

Takže napríklad pre DNF $\mathcal{F}_G$ z rovnice (3.1) máme $S(\mathcal{F}) = V, \forall e \in E : H_e(\mathcal{F}) = \emptyset, I(\mathcal{F}) = \emptyset, D(\mathcal{F}) = \emptyset, T(\mathcal{F}) = E$. Ak uvažujeme o procedúre forward chaining nad $\mathcal{F}$ so vstupnou množinou $\{v\}$, potom $S(\mathcal{F})$ označuje "štartovnú" (starting) množinu premenných, zahrnutých v procedúre do výslednej množiny ako prvé, množina $I(\mathcal{F})$ potom množinu "prostredných" (intermediate) a nakoniec $T(\mathcal{F})$ množinu "koncových" (terminal) premenných. Pri zachovaní tejto neformálnej terminológie nazvem termy objavujúce sa v Lemma 3.2 v kategórii a) ako štartovné, v kategórii b) ako prostredné a v kategórii c) ako koncové. Množina $H_e(\mathcal{F})$ je množina "hláv" (heads) prostredných termov odpovedajúcich hrane $e \in E$ a množina $D(\mathcal{F})$ potom odpovedá "doméne hláv" (domain of heads) prostredných termov.
V nasledujúcom Lemma ukážeme, že medzi všetkými minimálnymi (v počte termov, tak i ďalej v celej kapitole) DNF funkcie f_G je vždy aspoň jedna, ktorá má isté pekné vlastnosti v zmysle vyššie zavedeného značenia. Ba čo viac, takáto DNF sa dá získať z každej minimálnej DNF v polynomiálnom čase.

Lemma 3.3 *Z danej minimálnej DNF $\mathcal{F}_0$ funkcie f_G sa dá odvodiť iná minimálna DNF $\mathcal{F}_4$, ktorá je primárna a splňa nasledujúce podmienky:*

a) $S(\mathcal{F}_4)$ a $D(\mathcal{F}_4)$ sú disjunktné a pokrývajú celú V,

b) $H_e(\mathcal{F}_4) \cap H_{e'}(\mathcal{F}_4) = \emptyset$ pre všetky $e \neq e'$,

c) $I(\mathcal{F}_4)$ a $T(\mathcal{F}_4)$ sú disjunktné a pokrývajú celú E.

Dôkaz. Kompletný dôkaz sa dá nájsť v [4]. Na tomto mieste len podotknem, že tento dôkaz pozostáva zo sekvencie štyroch *prípustných modifikácií* pôvodnej DNF $\mathcal{F}_0$, ktoré na jednej strane zaručujú, že modifikovaná DNF bude ekvivalentná pôvodnej a na druhej strane zachovávajú počet termov (teda ak bola pôvodná DNF minimálna, tak aj modifikovaná bude minimálna).
□

Uvediem niekoľko ďalších vlastností DNF $\mathcal{F}_4$. Najprv zavediem značenie $N_e(\mathcal{F}_4) = \{e' \in E | e \neq e' \subseteq e \cup H_e(\mathcal{F}_4)\}$. Množina $N_e(\mathcal{F}_4)$ teda odkazuje na potenciálne termy, ktoré sa môžu použiť procedúrou forward chaining po tom, ako bola do výslednej množiny zaradená aj hrana e.
V ďalšom texte sa budeme zaoberať len DNF $\mathcal{F}_4$, preto na zjednodušenie značenia budem jej názov vynechávať na miestach, kde to bude možné. Teda napríklad namiesto $I(\mathcal{F}_4)$ budem písať len jednoducho I. Podobne pre ostatné značenia.

Lemma 3.4 *Pre každú hranu $e \in I$ platí $N_e \neq \emptyset$.*

Dôkaz. Vykonajme forward chaining nad $\mathcal{F}_4$ so vstupnou množinou $e \in I$. Po prvom kroku procedúry máme v $R(e)$ množinu $e \cup H_e$. Ak by teraz bolo $N_e = \emptyset$, potom by v tomto bode procedúra skončila bez toho, aby do $R(e)$ zaradila v. To by bol spor s faktom, že $e\bar{v}$ je

primárny implikant f_G.
□

Lemma 3.5 *Ak má graf G aspoň dve hrany (tj. $m > 1$), potom $I \neq \emptyset$.*

Dôkaz. Pre spor predpokladajme, že graf má hrany e, e' a zároveň $I = \emptyset$. Takže platí, že $e, e' \in T$, teda $\mathcal{F}_4$ obsahuje termy $e\bar{v}, e'\bar{v}$. Taktiež musí pre všetky $x \in V$ platiť, že $x \in S$. Teda $\mathcal{F}_4$ obsahuje aj termy $v\bar{x}$ pre všetky $x \in e' \setminus e$. Takže nasledujúcou modifikáciou získame DNF ekvivalentné $\mathcal{F}_4$ a obsahujúce o jeden term menej.

- Vymaž termy $e\bar{v}, v\bar{x}$ pre $x \in e' \setminus e$.
- Pridaj termy $e\bar{x}$ pre $x \in e' \setminus e$.

Na prvý pohľad je vidieť, že keď výsledok týchto modifikácií označíme ako $\mathcal{F}_5$, potom $\{v\} \in R_{\mathcal{F}_5}(e)$ a $\{x\} \in R_{\mathcal{F}_5}(v)$ pre všetky $x \in e' \setminus e$. Teda obe DNF sú skutočne ekvivalentné, čo je v spore s minimalitou $\mathcal{F}_4$.
□

Lemma 3.6 *Ak $m > 1$, potom existuje také $e \in I$, pre ktoré $e \subseteq S$.*

Dôkaz. Podľa Lemma 3.5 ak graf G obsahuje aspoň dve hrany, potom $I \neq \emptyset$, a teda aj $D \neq \emptyset$ (z definície). Nech $x \in D$ ľubovoľný, pozrime sa na $R(\{v\})$. Keďže $v\bar{x}$ je implikantom $f_G = \mathcal{F}_4$, musí platiť že $x \in R(\{v\})$. Nech $e\bar{y}$ je prvý prostredný term použitý v tomto forward chaining (takýto term musí existovať, keďže literál $\bar{x}$ sa objavuje v $\mathcal{F}_4$ iba raz, a to v prostrednom terme). To ale znamená, že vrcholy hrany e boli do $R(\{v\})$ zahrnuté použitím výlučne štartovných termov a teda $e \subseteq S$.
□

Lemma 3.7 *Ak $m > 1$, potom $|S| \geq 2, |I| \leq |D| \leq n-2$ a $\mathcal{F}_4$ obsahuje aspoň $m+2$ termov.*

Dôkaz. Prvá nerovnosť je priamym dôsledkom Lemma 3.6. Druhá nerovnosť je zas dôsledkom vlastnosti b) z Lemma 3.3. Nerovnosť $|D| \leq n-2$ vyplýva z Lemma 3.3 a) použitím nerovnosti $|S| \geq 2$. Na určenie počtu termov vo formuli si pripomeňme jej tvar:

$$\mathcal{F}_4 = (\bigvee_{x \in S} v\bar{x}) \vee (\bigvee_{e \in I} \bigvee_{x \in H_e} e\bar{x}) \vee (\bigvee_{e \in T} e\bar{v}).$$

Takže pre počet termov dostávame

$$\begin{aligned} \text{počet termov v } \mathcal{F}_4 &= |S| + \sum_{e \in I} |H_e| + |T| = \\ &= |S| + |D| + |T| \geq \\ &\geq |S| + |I| + |T| = |S| + m \geq \\ &\geq 2 + m. \end{aligned} \tag{3.3}$$

Prvá rovnosť vyplýva z Lemma 3.3 b). Následné nerovnosti vyplývajú priamo z nerovností ukázaných vyššie v tomto Lemma.
□

Teraz uvediem jednoduché ale užitočné Lemma, ktoré pojednáva špeciálny prípad, kedy $|I|$ nadobúda svoju maximálnu hodnotu, čiže $|I| = n - 2$.

Lemma 3.8 *Ak $|I| = n - 2$, potom*

a) $S = e$ pre nejaké $e \in E$;

b) $|H_e| = 1$ pre všetky $e \in I$;

c) pre všetky $e \in I$ a $e' \in N_e$ platí $|e \cap e'| = 1$, tj. hrany e a e' tvoria v G cestu dĺžky dva;

d) $\mathcal{F}_4$ obsahuje presne $m + 2$ termov.

Dôkaz.
Časť a).
Podľa Lemma 3.3 a) máme $|S| + |D| = n$ a podľa Lemma 3.7 máme $n-2 \geq |D| \geq |I| = n-2$, z čoho dostávame $|D| = n - 2$ a teda $|S| = 2$. Na druhej strane, podľa Lemma 3.6, S musí obsahovať hranu grafu G.
Časť b).
Podľa časti a) tohoto Lemma $|D| = |I|$ a teda použitím Lemma 3.3 b) dostávame, že $|H_e| = 1$ musí platiť pre všetky $e \in I$.
Časť c).
Toto je triviálny dôsledok časti b) tohoto Lemma.
Časť d).
Podľa časti b) máme $\sum_{e \in I} |H_e| = |I|$ a teda všetky nerovnosti v (3.3) sa stávajú rovnosťami.
□

Uvediem ešte posledné z technických Lemmat, ktoré boli v pôvodnom článku uvedené pred tým, ako sa pristúpilo k dokazovaniu Vety 3.1. Zároveň všetky tieto Lemma boli bezchybné a ukazovali pekné vlastnosti $\mathcal{F}_4$, ktoré využijem aj neskôr pri hľadaní problému ekvivalentnému hľadaniu minimálnej formy f_G.

Lemma 3.9 *Ak $\mathcal{F}_4$ obsahuje presne $m + 2$ termov, potom $|I| = n - 2$.*

Dôkaz.
Všetky nerovnosti v (3.3) sa v tomto prípade stávajú rovnosťami a to znamená, že $|S| = 2$ a $|I| = |D|$. Na základe Lemma 3.3 a) platí $|S| + |D| = n$ a teda $|I| = n - 2$.
□

Následne ako autor článku dokázal všetky tieto užitočné Lemmata, prešiel k dôkazu samotného tvrdenia. Avšak tu hneď zkraja došiel k chybnému predpokladu, že ku každému $e \in I$ existuje takzvaná e-**cesta**, tj. postupnosť hrán grafu G v tvare $\{e = e_1, e_2, \ldots, e_l\}$, kde pre každé $e_i, i = 1 \ldots l$ platí $e_i \subseteq E$. Naviac pre $e_j = (y_{j-1}, y_j)$ také, že $e_j \in N_{e_{j-1}}, j = 2 \ldots l$ (teda v $\mathcal{F}_4$ je term $e_{j-1}\bar{y}_j$ pre $j = 2 \ldots l$). Zároveň $e_l \in T$ (teda $\mathcal{F}_4$ obsahuje term $e_l\bar{v}$).
Toto sa zdalo byť ako celkom priamočiary dôsledok toho, že pre každé $e \in I : N_e \neq \emptyset$ a

zároveň $|H_e| = 1$ a naviac je toto H_e pre e unikátne (žiadne dve ho nemajú zhodné). A keďže naviac $|S| = 2$ a $S = e$ pre nejaké $e \in E$, bol to ideálny kandidát na začiatok takejto cesty. A vzhľadom k tomu, že žiadne dve hrany nemajú zhodné H_e a zároveň S a D tvoria množinu všetkých vrcholov sa zdalo, že to takmer nutne musí byť priamočiarou reprezentáciou Hamiltonovskej cesty. Avšak má to dva zádrhele:

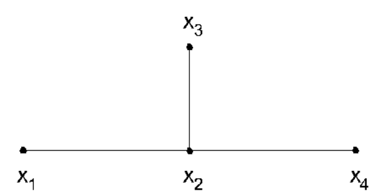
- $|H_e| = 1$, avšak N_e môže byť dvojprvkové - obe vrcholy hrany e môžu súčasne tvoriť "následníka"

- v e-ceste si vyberme $e_j = \{y_{j-1}, y_j\}, j \in \{2, \ldots, l\}$. Vieme, že $e_{j+1} \in N_{e_j}$. V článku sa automaticky predpokladá, že $e_{j+1} = \{y_j, y_{j+1}\}$. Avšak v skutočnosti sa môže veľmi ľahko stať, že $e_{j+1} = \{y_{j-1}, y_{j+1}\}$, teda že sa použije "predchádzajúci" vrchol ešte raz.

Prvý prípad, teda ten, kedy obe vrcholy môžu byť v N_e autor podrobne rieši. Je to zároveň moment, ktorý ten dôkaz pomerne značne komplikuje. Vytvára sa tým totiž akési "napojenie v strede" cesty. Tu zas príde vhod to, že graf G je kubický. To uľahčí situáciu v tom, že takéto napojenie sa v skutočnosti môže stať len na začiatku, alebo na konci e-cesty. Potom sa tieto dva prípady zvlášť riešia a príslušná DNF sa modifikuje tak, aby tieto napojenia realizovala, tj. predlžovala e-cestu až po moment, kedy bude obsahovať všetky vrcholy a stane sa tak cestou Hamiltonovskou.

Druhý prípad však nebol v dôkaze braný v zreteľ. A ako ukážem v nasledujúcej časti tejto kapitoly, mal ďalekosiahle následky v tom, že výsledná e-cesta sa už ani zďaleka nepodobala na cestu Hamiltonovskú.

3.2 Skutočná tvár f_G



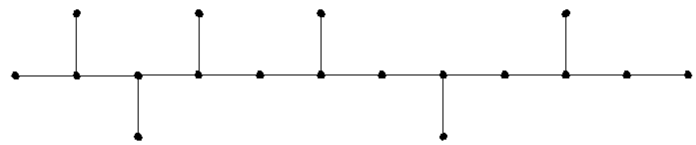
Obr. 3.1: Najjednoduchší protipríklad

Na obrázku 3.1 je hneď prvý najjednoduchší protipríklad aký ma napadol. Zobral som si DNF tvaru:

$$x_1x_2\bar{x_3} \vee x_2x_3\bar{x_4} \vee x_2x_4\bar{v}$$

Kde práve posledný term narušuje pôvodnú predstavu e-cesty. Na základe definície e-cesty by namiesto neho mal byť správne term $x_3x_4\bar{v}$. Tým by sa nevytvorila "odbočka", ale cesta by šla ďalej rovno.

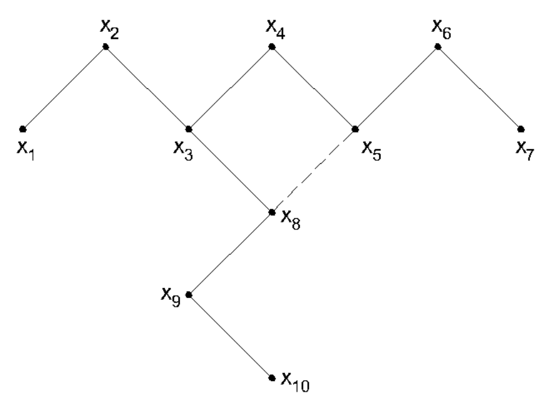
Takže z toho vyplývalo, že namiesto rovnej cesty, sa vytvárala akási "stonožka" - cesta s krátkymi odbočkami tak, ako je naznačená na obrázku 3.2. Povedal som si, že to by ale obtiažnosti problému nemuselo vadiť. Veď nájsť takúto jednu dlhú stonožku v grafe



Obr. 3.2: Stonožka vytvorená namiesto rovnej cesty

prechádzajúcu všetkými vrcholmi je akiste rovnako ťažké ako nájsť cestu. Tento prevod nie je dotiahnutý, ale len pre predstavu. Majme graf a chceme v ňom nájsť Hamiltonovskú cestu. Ku každému vrcholu v grafe stupňa tri a viac pridáme ešte jeden vrchol navyše a spojíme ich hranou. Potom by nájdenie stonožky v tomto grafe odpovedalo nájdeniu Hamiltonovskej kružnice v pôvodnom.

Táto myšlienka vychádzala z toho, že napojovanie jednotlivých stonožiek na seba sa riadi rovnakými pravidlami ako napojovanie pôvodných e-ciest. A síce, že napojenie je možné uskutočniť len na konci alebo na začiatku. V pôvodnom dôkaze to bolo takto preto, lebo uprostred cesty má každý vrchol stupeň 2. Na napojenie je však potrebná hrana mimo túto cestu. Ale takáto hrana by automaticky zvýšila stupeň oboch participujúcich vrcholov na tri, a teda by už nezostala “voľný slot” pre napojenie odbočky. Z toho potom vyplývalo, že sa odbočka môže uskutočniť len na konci alebo na začiatku - keďže jedine tam sú vrcholy stupňa jedna. Lenže v tejto stonožke sú, ako vidieť, vrcholy stupňa jedna aj uprostred - to sú tie “nohy” stonožky. Takže napojenie sa dá uskutočniť aj na každej takejto nohe - viď obrázok 3.3.



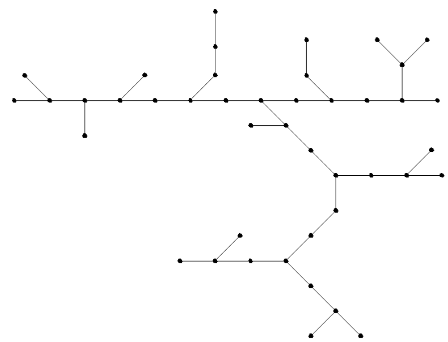
Obr. 3.3: Napojenie jednej stonožky v prostriedku inej stonožky.

Na tomto obrázku je nakreslený graf generovaný nasledujúcou DNF:

$$(x_1x_2\bar{x_3} \vee x_2x_3\bar{x_8} \vee x_3x_8barx_4 \vee x_3x_4\bar{x_5} \vee x_4x_5\bar{x_6} \vee x_5x_6\bar{x_7} \vee x_6x_7\bar{v}) \vee (x_5x_8\bar{x_9} \vee x_8x_9\bar{x_{10}} \vee x_9x_{10}\bar{v})$$

Kde v zátvorkách sú jednotlivé stonožky. Prvá zátvorka odpovedá vodorovnej stonožke, druhá zátvorka potom vertikálnej stonožke, ktorá sa na tú horizontálnu napojí pomocou hrany $\{x_5, x_8\}$, ktorá leží mimo e-cestu.

No dobre, tak skúsme si predstaviť, ako by vyzeralo viacnásobné napojenie takých stonožiek. Na obrázku 3.4 je viacero stonožiek na seba napojených.



Obr. 3.4: Viacnásobné vzájomné napojenie stonožiek.

Nuž ale to je vlastne presne kostra grafu. Je vidieť na prvý pohľad, že každá kostra grafu sa dá takto vyskladať zo stonožiek. Zdalo sa teda, že ľubovoľná kostra grafu generuje DNF s minimálnym počtom termov. Zdalo sa to prirodzené aj vďaka tomu, že ľubovoľná kostra grafu a Hamiltonovská kružnica zdieľajú istý spoločný parameter "minimality" - a to, že majú $|E| = |V| - 1$. Čo vlastne odpovedá jednotlivým termom v minimálnej DNF - pre každú hranu jeden term (mimo počiatočnej množiny S). Ak by bola toto pravda, tak by bol pôvodný dôkaz úplne nepoužiteľný.

3.2.1 Stonožkovanie

Keď som sa však pokúsil vytvoriť algoritmu, ktorý by z ľubovoľnej kostry generoval minimálnu DNF, veľmi rýchlo som narazil na problém. A tým je práve tá "napojujúca hrana", ktorá je mimo tejto cesty. Na obrázku 3.3 je to hrana $\{x_5, x_8\}$. Ako sa ukázalo, táto hrana tento problém značne komplikovala. Pre každé napojenie sa totiž musela nájsť hrana. A to z podgrafu, ktorý už bol vytvorený. To ma viedlo k definícii Stonožkovania, tj. pokrytia grafu G stonožkami tak, aby som mal tieto spätné hrany. Prirodzeným požiadavkom teda bolo mať jednu Hlavnú Stonožku, z ktorej by sa vytvárali ostatné. Stonožkovanie som nadefinoval nasledovne:

Definícia 3.10 *Podgraf S grafu G sa nazýva **stonožkou v grafe G** ak sa množina vrcholov grafu S dá rozdeliť na dve disjunktné množiny $T(S)$ (vrcholy tela) a $N(S)$ (vrcholy nôh) tak, aby platilo:*

a) množina všetkých vrcholov z $T(S)$ tvorí cestu

b) $\forall v \in T(S)$: všetci susedia vrcholu v (v grafe G) sú v $N(S)$

c) $T(S)$ tvorí cestu. Obe koncové body tejto cesty sú stupňa 1 (tzv. konce tela*).*

Definícia 3.11 *Kostra K grafu G sa nazýva **stonožkovaním**, ak obsahuje aspoň jednu stonožku a zároveň je možné zvoliť si jednu stonožku (nazveme ju Hlavná stonožka - HS) tak, aby boli splnené nasledujúce podmienky:*

a) Konce nôh hlavnej stonožky sú hranou grafu G spojené s nejakým vrcholom jej tela (táto hrana však nie je súčasťou kostry, čo je prirodzené keďže kostra neobsahuje cykly). Čiže je splnená nasledujúca podmienka:

$$\forall v \in N(HS) \exists e = \{v, y\} \in E(G \setminus K) : y \in T(HS)$$

b) Každý vrchol K je súčasťou nejakej stonožky v K

c) Každý vrchol $v \in T(S)$ pre nejakú stonožku S, nie je prvkom inej stonožky než S

d) Každá stonožka v K je ukotvená v K

Definícia 3.12 *Nech K je stonožkovaním grafu G. Stonožka S je* ***ukotvená v K*** *ak:*

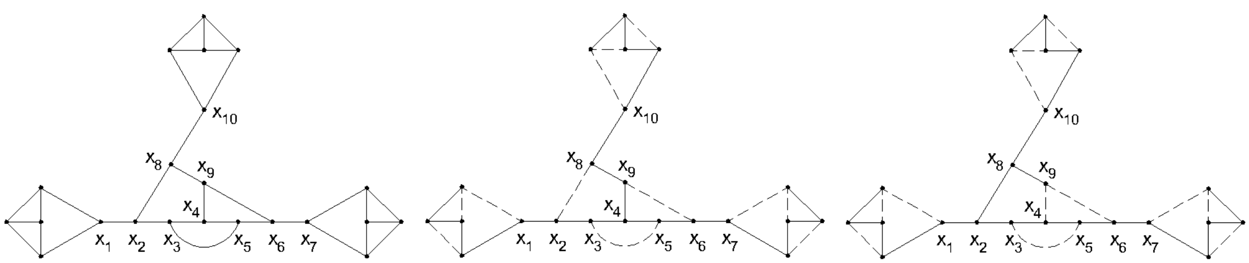
a) Je Hlavnou stonožkou K

b) Je napojená na nejakú stonožku ukotvenú v K

Definícia 3.13 *Nech K je stonožkovaním grafu G. Stonožka S je* ***napojená*** *na stonožku T v K ak:*

a) Pre nejaký koniec tela *v stonožky S platí* $v \in N(T)$

b) V grafe $G \setminus K$ *existuje hrana* $\{v, y\}$, *kde* $y \in T(U)$ *pre nejakú stonožku U ukotvenú v K*



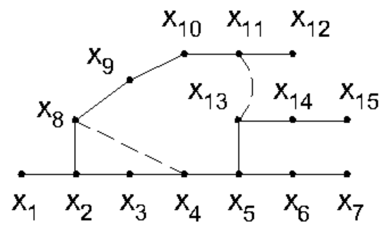
(a) Kubický graf (b) Stonožkovanie grafu (c) Nestonožková kostra

Obr. 3.5: Kubický graf a jeho kostry. Jedna je stonožkovaním, druhá nie je.

Teraz vyvstala otázka, či môže existovať graf v ktorom existuje stonožka a zároveň v ktorom je kostra, ktorá stonožkou nie je. Pokúsil som sa takýto graf nájsť. A skutočne sa mi to podarilo. Na obrázku 3.5 je tento kubický graf a jeho dve kostry. Jedna z týchto kostier je stonožkovaním, druhá nie je. (To, že jedna je stonožkovaním je očividné. To, že druhá nie je by sa dalo ukázať rozborom všetkých možných Hlavných Stonožiek. Týchto možností je v skutočnosti len tri.)

Nuž ale toto situáciu komplikovalo. Vyzeralo to, že problém hľadania stonožkovania bude vskutku NP-ťažký, keďže budem musieť zrejme prehľadávať všetky kostry a u každej z nich zisťovať, či je alebo nie je stonožkovaním.

V snahe dokázať ekvivalenciu medzi hľadaním minimálnej DNF a hľadaním stonožkovania v grafe som dospel k jednému príkladu, ktorý ju vyvrátil. Na obrázku 3.6 vidíme kostru grafu, ktorá generuje minimálnu DNF ale nie je stonožkovaním. Tento protipríklad ma už potom priamo viedol na stopu definitívnemu riešeniu - problému HV-pokrytia. Ako ukážem v nasledujúcej sekcii, problém nájdenia HV-pokrytia je skutočne ekvivalentný hľadaniu minimálnej DNF funkcie f_G.



Obr. 3.6: Kostra grafu generujúca minimálnu DNF funkcie f_G, ktorá nie je stonožkovaním.

3.3 HV-pokrytie

V tejto časti uvediem vlastný výsledok tejto kapitoly. Tým bude nadefinovanie nového problému HV-pokrytia, ktorý je ekvivalentný hľadaniu minimálnej reprezentácie funkcie f_G. Pre túto ekvivalenciu týchto problémov uvediem aj podrobný dôkaz vo forme algoritmov na prevod, včetne dôkazov ich korektnosti a zložitosti.
V priebehu vypracovávania práce sa mi ani po nemalej vynaloženej námahe nepodarilo nájsť polynomiálny algoritmus na hľadanie tohto HV-pokrytia. V článku [4], ktorý som rozoberal v tejto kapitole, bolo ukázaných nemálo užitočných a nie úplne jednoduchých Lemmat, ktoré ukazovali pekné vlastnosti a vyzerali byť pomerne užitočné. Tvar funkcie f_G sa mi zdal dostatočne všeobecný. Všetky tieto fakty dohromady vo mne evokovali pocit, že problém nájdenia HV-pokrytia bude vskutku NP-ťažký. Tento fakt sa mi bohužiaľ nepodarilo ukázať. Uvediem tu však aspoň čiastočný cieľ - ukázal som, že problém hľadania HV-pokrytia je ekvivalentný hľadaniu minimálnej reprezentácie funkcie f_G. Toto tvrdenie môže neskôr slúžiť na opravu pôvodného dôkazu.

3.3.1 Definícia

Pri špecifikovaní presnej definície som vychádzal z protipríkladu pre stonožkovanie (obrázok 3.6), pričom som sa snažil odstrániť nedostatok stonožkovania. Ten spočíval v tom, že netrebalo, aby do každej odbočky viedla napojujúca hrana z Hlavnej stonožky, ale aby do každej odbočky viedla hrana z "už vygenerovaného grafu". To znamená, že začnem v hrane, ktorú v $\mathcal{F}_4$ reprezentuje množina S, a potom pôjdem po jednej ceste k ľubovoľnému listu. Túto cestu nazvem hlavná, všetky vrcholy na nej sú H. Ak budem chcieť odbočiť na vedľajšiu cestu (do vrchola s označením V), budem musieť mať okrem samotnej odbočky aj napojujúcu hranu z nejakého vrcholu, v ktorom som už bol.

Takže v obrázku 3.6 je zároveň zobrazený jeden modelový príklad HV-pokrytia, kde $v_S = x_1, f(x_8) = V, f(x_{13}) = V$, ostatné vrcholy majú ohodnotenie H.

Definícia 3.14 ***HV-pokrytie*** *kubického grafu G je trojica (K, f, v_S), kde K je kostra grafu G, $v_S \in G$ je vrchol v grafe, f je funkcia $f : V(K) \to \{H, V\}$ splňujúce nasledujúce podmienky:*

a) v_S je v K list ($\deg_K(v_S) = 1$)

b) $f(v_S) = H$

c) $\deg_K(v) = 3 \Rightarrow f(v) = H$

d) ak v nesusedí so žiadnym vrcholom stupňa 3, potom $f(v) = H$

e) pre každý vrchol v stupňa 3 v K ($\deg_K(v) = 3$) platí:

- *nech u je ten jeho sused, ktorý je najbližšie k v_S v K. Potom $f(u) = H$.*
- *spomedzi zvyšných dvoch jeho susedov má 1 hodnotu f rovnú H, druhý V*

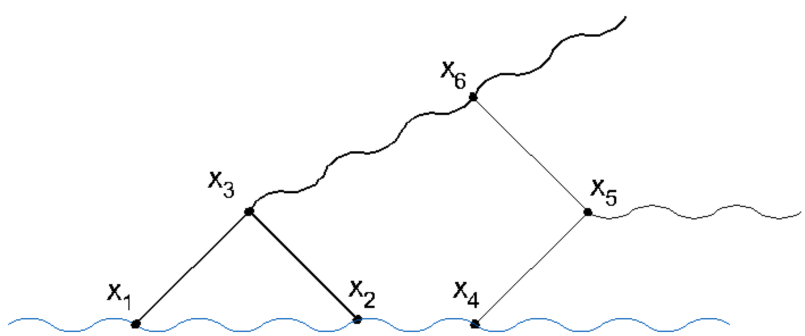
f) do každého vrcholu grafu vedie v (K, f) HV-cesta z vrcholu v_S

Definícia 3.15 *Majme kubický graf G, jeho kostru K, funkciu $f : V(K) \to \{H, V\}$. Z vrcholu u vedie do vrcholu v* ***HV-cesta*** *v (K, f) ak:*

a) $u = v$ (HV-cesta je reflexívna)

b) $\exists w : \{v, w\} \in E(K)$ a existuje HV-cesta z u do w

c) ak $f(v) = V \wedge \deg_K(v) > 1$, potom musí byť naviac splené:
$\exists w : \{v, w\} \in E(G \setminus K)$ a existuje HV-cesta z u do w.



Obr. 3.7: Graf s odbočkami - pre každú dve napojujúce hrany.

Predstavme si jednu odbočku - tú predstavuje vrchol s ohodnotením V - je to vrchol, do ktorého ide hrana v príslušnej kostre a zároveň do neho ide hrana mimo túto kostru. Takže v pôvodnom grafe je tento vrchol spojený dvoma rovnocennými hranami s už odvodeným podgrafom. Túto situáciu znázorňuje obrázok 3.7 (kde $f(x_3) = V, f(x_5) = V$). Je vidieť, že z týchto dvoch hrán, ktorými je odbočovací vrchol spojený s pôvodným grafom, si môžem ľubovoľnú z nich zvoliť za hranu v HV-pokrytí a tú druhú za napojujúcu hranu. Z toho je vidieť, že ak mám v HV-pokrytí k vrcholov stupňa 3, potom existuje 2^k rôznych ekvivalentných HV-pokrytí, ktoré sa dajú nájsť v polynomiálnom čase (avšak očividne môže byť ich počet až exponenciálny).

Poznámka: Ak bude jasné o aké K a aké f uvedené v definícii sa jedná, budem pre jednoduchosť písať iba "HV-cesta".

V definícii HV-pokrytia, v bode e) som spomenul suseda, korý je "najbližšie k v_S v K". Taký sused vždy existuje a je práve jeden. K je kostra, takže pre každé dva vrcholy existuje práve jedna cesta, ktorá ich spája. Tak existuje aj cesta, ktorá spája vrcholy v_S a v. Potom toto u je ten jeho sused, ktorý sa v tejto ceste nachádza (nachádzať sa tam aspoň jeden musí (keďže $\deg_K(v_S) = 1$) a je to maximálne jeden, lebo inak by sa vytvoril v K cyklus).
Teraz prejdeme k Vete vyjadrujúcej ekvivalenciu problému HV-pokrytia s hľadaním minimálnej DNF funkcie f_G.

Veta 3.16 *Majme kubický graf G. Jemu odpovedajúca čiste Hornovská f_G má DNF reprezentáciu s maximálne $m + 2$ termami práve vtedy ak existuje HV-pokrytie grafu G.*

Dôkaz uvediem vo forme algoritmov na prevod problémov. Najprv z HV-pokrytia na DNF f_G, potom opačne.
Poznámka: Ak budem na nejakom mieste dôkazu písať symbol typu x_i, budem odkazovať na premennú odpovedajúcu nejakému vrcholu. Ak použijem symbol typu y_i budem odkazovať na hocijakú premennú, tj. buď premennú odpovedajúcu nejakému vrcholu, alebo na premennú v.

3.3.2 HV-pokrytie → DNF

Majme teda HV-pokrytie (K, f, v_S) grafu G. Ja z neho skonštruujem DNF $\mathcal{F}$, ktorá bude reprezentáciou funkcie f_G a bude mať maximálne $m + 2$ termov.

V algoritme používam tieto premenné, popíšem ich podľa toho, ako vystupujú v kroku2:

- x Premenná, ktorá sa práve spracováva. Do $\mathcal{F}$ pridám termy tvaru $ax\bar{b}$, kde a je jej už spracovaný sused, b sú všetci jej ostatní susedia.
- p Už spracovaný predchodca premennej x.
- $\mathcal{F}$ Tvoriaca sa výsledná DNF.
- O Fronta s odbočkami - vrcholmi označenými ako V, na ktoré sa dostane rad až sa nájde napojujúca hrana.

Popis algoritmu:

1) Inicializácia.
   Nech x je (ten jediný) sused vrcholu v_S v K.

$\mathcal{F} := \{v\bar{v_S}, v\bar{x}\}, p := v_S, O := \emptyset$, všetky vrcholy grafu označ ako nespracované, vrchol v_S označ ako spracovaný. Všetky hrany označ ako nespracované.
Pokračuj krokom 2).

2) Idem po hlavnej,
a to od vrcholu x, smerom k listom.
Označ vrchol x ako spracovaný, označ hranu $\{p, x\}$ ako spracovanú. Pozri sa, koľkých nespracovaných susedov má x v K:

   (a) Žiadneho. Som v liste. Teda na konci cesty.
   $\mathcal{F} := \mathcal{F} + px\bar{v}$
   Ak je O prázdna, pokračuj krokom 4).
   Inak pokračuj krokom 3).

   (b) Jedného, označme ho x_1. Jednoduchá rovná cesta.
   $\mathcal{F} := \mathcal{F} + px\bar{x}_1, p := x, x := x_1$
   Pokračuj krokom 2).

   (c) Dvoch, označme ich x_1, x_2 tak, aby platilo: $f(x_1) = V, f(x_2) = H$. Odbočka.
   $\mathcal{F} := \mathcal{F} + \{px\bar{x}_1, xx_1\bar{x}_2\}, p := x, x := x_2$,označ hranu $\{x, x_1\}$ ako spracovanú.

      - $\deg_K(x_1) = 1$
      Označ x_1 ako spracovaný.
      - $\deg_K(x_1) = 2$
      $O := O + x_1$

   Pokračuj krokom 2).

3) Nájdi schodnú odbočku.
Nájdi hranu $\{x_1, x_2\} \in E(G \setminus K)$ takú, že x_1 je spracovaný a x_2 je v O.
$p := x_1, x := x_2, x_2$ vyber z O
Pokračuj krokom 2).

4) Nespracované hrany.
Pre každú nespracovanú hranu $\{x_1, x_2\} \in E(G)$:
$\mathcal{F} := \mathcal{F} + x_1x_2\bar{v}$, označ ju ako spracovanú.

Korektnosť kroku 2).
Píše sa tam "označ hranu $\{p, x\}$ ako spracovanú". Existuje takáto hrana? Overím si prípady, kedy do x niečo vkladám. V kroku 1) priamo píšem "nech x je sused vrcholu v_S", takže táto hrana existuje. Vo všetkých ostatných prípadoch keď do x niečo vkladám, tak napĺňam aj premennú p. A to zakaždým tak, aby boli susedia.
To, prečo si v kroku 2)a môžem dovoliť ukončiť algoritmus, odvodím nižšie v časti venovanej úplnosti algoritmu.
V kroku 2)c je x stupňa 3, takže môžem bez ujmy predpokladať, že jeden z jeho susedov je H, druhý V.

Korektnosť kroku 3).
Zavediem si alternatívnu definíciu HV-pokrytia. Tá nech je definovaná presne rovnako ako riadna, ale s tým rozdielom, že namiesto HV-cesty budem požadovať alternatívnu HV-cestu. A tá bude opäť definovaná presne rovnako ako riadna, ale s tým rozdielom, že bod c) nahradím nasledujúcim textom:

c) ak $f(v) = V \wedge \deg_K(v) > 1$, potom neexistuje HV-cesta z u do v.

Ak prídem do tohto kroku algoritmu, tak označené ako spracované sú presne tie vrcholy, ktoré by splňovali alternatívnu definíciu HV-pokrytia. Ale žiaden z nespracovaných túto alternatívnu definíciu nesplĺňa (včetne tých, čo sú vo fronte O).
V oboch definíciách je podmienka, že príslušný vrchol musí byť hranou z K spojený s nejakým už spracovaným vrcholom (keďže spracované sú presne tie, ktoré v tomto kroku splĺňajú definíciu existencie HV-cesty, môžem si dovoliť tieto dva pojmy zamieňať). Ale kde sa môže nejaký nespracovaný vrchol hranou napojiť na nejaký spracovaný? Keď sa pozriem na priebeh algoritmu, tak zistím, že som začal so susedom v_S a potom som postupoval tak, že vždy som každého suseda buď napojil, alebo dal do fronty O. Z toho vyplýva, že práve jedine vrcholy, ktoré sú v O sa môžu napojiť. Takže ak mám ešte nejaký nespracovaný vrchol zaradiť medzi spracované, musím najprv zaradiť aspoň jeden vrchol spomedzi O. Ale pre všetky tieto vrcholy vieme, že splňujú podmienky HV-pokrytia (predpoklad), ale nesplňujú podmienky alternatívneho HV-pokrytia. Z toho vyplýva, že musí spomedzi vrcholov v O existovať aspoň jeden taký, pre ktorý je podmienka c) splená. To som mal ukázať.

Konečnosť algoritmu.
V každom kroku označím minimálne jeden vrchol ako spracovaný. Ak sú všetky vrcholy spracované, algoritmus končí. Vrcholov je konečný počet.

Úplnosť algoritmu.
V tomto odstavci ukážem, že algoritmus označil všetky vrcholy ako spracované. Na základe toho potom nižšie dokážem (v odstavci "Odpovedá $\mathcal{F}$ funkcii f_G?"), že takto vytvorená DNF reprezentuje celú f_G.
Nech teda existuje vrchol v taký, že prebehnutí algoritmu nebol zaradený medzi spracované. Zoberme si cestu $P = (v_S = v_1, v_2, v_3, \dots v)$ v K z vrcholu v_S do v (z definície kostry grafu takáto cesta existuje). Vieme, že v_S je označený ako spracovaný (v kroku 1)) a v nespracovaný. Označme teda v_i prvý nespracovaný vrchol na tejto ceste. Z toho vyplýva, že $v_j = v_{i-1}$ je spracovaný a sú spojení navzájom hranou. Ak $f(v_i) = H$, potom bol pri spracovávaní vrcholu v_j určite vložený do premennej x a teda v nasledujúcom kroku spracovaný. Ak $f(v_i) = V, \deg_K(v_i) = 1$, potom bol v kroku 2)c pri spracovávaní v_j okamžite označený tiež ako spracovaný. A konečne ak $f(v_i) = V, \deg_K(v_i) = 2$, potom bol pri spracovávaní v_j v kroku 2)c vložený do fronty O. Takže fronta O bola neprázdna. No ale pri dokazovaní korektnosti kroku 3) som ukázal, že vždy keď je O neprázdna, tak sa tam nájde jeden taký vrchol, ktorý zaradím medzi spracované. No ale vždy v kroku 3) jeden prvok z O vyberiem a spracujem. Prvkov je konečný počet, takže ak by som aj vždy vyberal z O iný prvok ako v_i, v konečnom počte cyklov by som sa dostal do stavu, kedy je vo fronte O iba v_i. Ale aj pre túto konštaláciu fronty platí vyššie uvedené tvrdenie, takže nutne musí byť v_i zaradené medzi spracované.

Zložitosť algoritmu.
Krok 1) trvá $O(m)$ - inicializácia datových štruktúr.
Pri vhodne zvolených datových štruktúrach trvá krok 2) konštantný čas.
V kroku 3) prechádzam frontou O a pre každý prvok sa v konštantnom čase pozriem na vrcholy s ktorými je spojený. Trvá teda maximálne $O(n)$.
Krok 4) trvá $O(m)$.

V každom cykle algoritmu môžem prejsť krokmi 2) a 3), jeden krok trvá teda čas maximálne $O(n)$. Ako som spimínal v časti venovanej Konečnosti algoritmu, počet cyklov je maximálne n.
Celková zložitosť algoritmu je teda $O(m + n^2)$.

Korektnosť algoritmu.
Odvodením korektnosti jednotlivých krokov sme ukázali, že algoritmus môže vždy vykonať jednotlivé kroky. Konečnosťou zase, že skončí v reálnom čase. Takže nám v konečnom čase algoritmus vydá DNF $\mathcal{F}$. Musíme ešte overiť podmienky, ktoré sa na ňu kladú. A síce, že reprezentuje f_G a že má maximálne $m + 2$ termov. Tým bude dôkaz hotový.
Najprv si však odvodím jedno pomocné tvrdenie, ktoré je nevyhnutné k dokázaniu týchto dvoch podmienok.

Lemma 3.17 *$\mathcal{F}$ splĺňa všetky vlastnosti kladené na $\mathcal{F}_4$ v znení Lemma 3.3.*

Týmto bude dokázané, že splĺňa aj všetky nasledujúce vlastnosti odvodené z tohto tvaru.
Dôkaz.

- $\mathcal{F}$ je primárna.
  Z Lemma 3.2 vieme, že každý term, ktorý je tvaru a) b) alebo c) je primárnym implikantom. V kroku 1) som pridával len termy tvaru a). V kroku 2)a zas len termy tvaru c) (už vyššie som ukázal, že v tomto kroku vždy platilo, že $\{p, x\}$ je hrana). V kroku 2)b len termy tvaru b), keďže $\{p, x\}$ je hrana a x_1 je rôzny od oboch (nie je x, lebo je jeho susedom a nie je p, lebo p je spracovaný a x_1 nie je). No a konečne v kroku 2)c opäť len termy tvaru b) - odôvodnenie rovnaké ako v predchádzajúcom prípade. V kroku 3) som do $\mathcal{F}$ nič nepridával a to, že všetky termy pridané v kroku 4) sú tvaru c) je očividné.

- Vlastnosť a).
  Množina S je tvorená vcholom v_S a jeho susedom. Žiaden z týchto vrcholov už nepridám ako negatívne premenné (očividné - p sa nikde nepridáva ako negatívna premenná).
  Teraz ukážem, že žiadnu premennú nepridám dvakrát ako negatívnu. Termy, ktoré nás zaujímajú sa pridávajú jedine v kroku 2)b a 2)c. Dá sa ľahko vypozorovať, že keď raz premennú označím ako spracovanú, nikdy sa už nedostane do nejakého termu ako negatívna (vždy pridávam ako negatívne premenné len susedov práve spracovávanej premennej). Pozrime sa na jednotlivé kroky, či sa nemôže stať, že nejakú premennú pridám dvakrát. V kroku 2)b pridávam premennú x_1. Ale hneď nato ju vložím do premennej x a teda hneď na začiatku nasledujúceho cyklu ju označím ako spracovanú (a teda sa už nikdy nevloží ako negatívna). V kroku 2)c čaká premennú x_2 rovnaký osud. Čo sa deje s premennou x_1, ktorú sme tiež pridali ako negatívnu? Ak $\deg_K(x_1) = 1$, potom ju označím hneď ako spracovanú a teda je všetko v poriadku. V tom druhom prípade ju vložím do fronty O. Z tejto fronty sa vyberajú prvky len v kroku 3). A tam sa táto premenná vkladá hneď do premennej x, tok algoritmu sa pošle na krok 2), v ktorom sa hneď na začiatku označí ako spracovaná.
  Popočítam, koľko termov tvaru $x_i x_j \bar{x_k}$ pridám. Ak ich bude $n - 2$, ukázal som teda, čo som mal. Vieme, že každý vrchol sa spracuje práve raz. Pozrime sa, koľko termov nami požadovaného tvaru sa do $\mathcal{F}$ vkladá pri jednom takomto spracovávaní v kroku 2).

- Ak je vrchol stupňa 1, nevloží sa ani jeden.
- Ak je vrchol stupňa 2, vloží sa práve jeden.
- A ak je vrchol stupňa 3, vložia sa 2 takéto termy.

Jediný vrchol, ktorý sa nespracováva v kroku 2) je vrchol v_S. Ten je ale stupňa 1 a nepridáva žiaden term nami požadovaného tvaru, takže vyššie uvedené platí pre všetky vrcholy.
Zavediem si teraz označenie. Nech n_1 = počet vrcholov stupňa 1, n_2 = počet vrcholov stupňa 2 a n_3 = počet vrcholov stupňa 3.
V teórii grafov sa dá veľmi ľahko odvodiť, že pre každý kubický graf, ktorý je stromom platí, že $n_3 = n_1 - 2$. (Pre cestu to platí. Vždy keď na ňu v strede napojím nejakú ďalšiu cestu, zvýšim počet listov o jedničku aj počet vrcholov stupňa 3 o jedničku. Keď napojím na list, nezvýšim ani jedno. Každý strom sa dá takto vyskladať z ciest.)
Z rovníc

$$n_1 + n_2 + n_3 = n$$

$$n_3 = n_1 - 2$$

dostávam

$$n_2 = n + 2 - 2 * n_1.$$

Spočítam teda počet termov, ktoré nás zaujímajú

$$\sharp \text{ termov} = 0*(n_1)+1*(n+2-2*n_1)+2*(n_1-2) = n+2-2*n_1+2*n_1-4 = n-2$$

- Vlastnosť b).
  Už v predchádzajúcom kroku som ukázal, že žiadnu premennú nepridám dvakrát ako negatívnu.

- Vlastnosť c).
  Vždy keď som v algoritme vložil do $\mathcal{F}$ nejaký term tvaru $x_1 x_2 \bar{y}, y \in V(G) \cup v$, označil som hranu $\{x_1, x_2\}$ ako spracovanú. Mohol som pridať nejaký term tvaru $x_i x_j \bar{y}$ kde by platilo že $\{x_i, x_j\}$ je už spracovaná? Keď algoritmus vstupuje do kroku 2) platí, že hrana $\{p, x\}$ ešte nie je spracovaná. Ukážem to rozborom prípadov podľa toho, odkiaľ môžem do kroku 2) vstúpiť.
  Keď algoritmus vchádza do kroku 2) z kroku 1), všetky hrany sú označené ako nespracované.
  Platí: každá spracovaná hrana spojuje len dva spracované vrcholy, alebo jeden spracovaný a druhý prvkom O (to vyplýva z toho, že vždy keď označujem nejakú hranu ako spracovanú, tak toto tvrdenie platí). Takže keď do kroku 2) vchádza z kroku 2)b, je $x := x_1$, ale toto x_1 je nespracované (podľa toho som ho vyberal), takže je nespracovaná aj hrana $\{x, x_1\}$, ktorá bude pri následnom vstupe do bodu 2) hranou $\{p, x\}$.
  Keď do kroku 2) vchádzam z kroku 2)c, stane sa budúcou hranou $\{p, x\}$ práve hrana $\{x, x_2\}$. Ale v tomto prípade je x_2 nespracovaný, takže ani táto hrana nemôže byť spracovaná.
  A konečne keď do kroku 2) vchádzam z kroku 3), tak sa skúmanou hranou stane $\{x_1, x_2\}$. Táto hrana však nemôže byť spracovaná, pretože ju vyberám z $E(G \setminus K)$. A jediné miesto, v ktorom vyberám hrany z tejto množiny je práve len krok 3). Ale ak v

tomto kroku raz nejakú hranu vyberiem, tak hneď v nasledujúcom cykle sa z nej stane spracovaná (a aj oba jej vrcholy, čím je jej opätovné vybratie v tomto kroku úplne nemožné).
Z tohto všetkého teda viem, že žiadnu hranu som nepridal dvakrát (či už bola ako negatívna premenná v, alebo nejaký vrchol grafu). Takže vieme, že $I(\mathcal{F}) \cap T(\mathcal{F}) = \emptyset$.
Môže sa stať, aby existovala po skončení algoritmu v G nejaká nespracovaná hrana? Vďaka kroku 4) je to nemožné.

□

Má $\mathcal{F}$ presne $m+2$ termov?
Keďže $\mathcal{F}$ je tvaru $\mathcal{F}_4$, platí pre neho aj Lemma 3.8 a stačí mi teda ukázať, že $|I| = n-2$. Ale to vyplýva priamo z pred chvíľou dokázaných vlastností a) a b) a z faktu, že $|S(\mathcal{F})| = 2$ (keďže termy tohoto typu pridávam len v kroku 1) algoritmu a tam ich pridám dva).

Odpovedá $\mathcal{F}$ funkcii f_G?

- $\mathcal{F} \leq f_G$
  Mám ukázať, že každý term z $\mathcal{F}$ je implikantom funkcie f_G. V dôkaze Lemma 3.17 som dokonca ukázal, že každý term z $\mathcal{F}$ je primárnym implikantom funkcie f_G.

- $f_G \leq \mathcal{F}$
  Vieme, že $f_G = \left(\bigvee_{x_i \in V} v\bar{x}_i\right) \vee \left(\bigvee_{(x_i,x_j)\in E} x_i x_j \bar{v}\right)$. Takže dôkaz si rozdelíme na dve časti.

  – $v\bar{x}_i \leq \mathcal{F}, \forall x_i \in V$
    Ukážem, že keď procedúra forward chaining začne s premennou v, tak vo výslednej množine procedúry budú všetky vrcholy grafu. Tým to bude dokázané. (Algoritmus sám vlastne úplne presne kopíruje činnosť procedúry forward chaining, ale skúsim to ukázať precízne.)
    Z Lemma 3.17 vieme, že pre každý vrchol x existuje v $\mathcal{F}$ term, v ktorom tento vrchol vystupuje ako negatívna premenná. A opäť - ak ukážem, že vždy keď takýto term pridávam, tak jej pozitívne premenné sú odvoditeľné z v, tak som hotový.
    V kroku 1) to platí. Keď prídem z kroku 1) do kroku 2), tak v ten moment sú obe premenné p a x odvoditeľné z v. Takže keď potom v bode 2)b a 2)c pridávam term $px\bar{x}_1$, tak pre tento term tvrdenie platí. V kroku 2)c pridávam navyše aj term $xx_1\bar{x}_2$. Ale pre tento term to tiež platí, keďže som súčasne s ním vložil aj term $px\bar{x}_1$. V oboch prípadoch potom pokračujem opäť do bodu 2), ale s novými premennými p a x. Ale tie sú vzhľadom k práve pridaným termom tiež odvoditeľné z v.
    Posledným prípadom zostáva, keď do kroku 2) prídem z kroku 3). Tam som premennú x_2 vybral z O. Ale keď si povšimneme situáciu, v ktorej sa do O vložila, zistíme že v ten moment bola odvoditeľná z v. Ďalšiu premennú volím spomedzi spracovaných. Ale vždy keď označujem premennú ako spracovanú, tak je taktiež odvoditeľná z v.

  – $x_i x_j \bar{v} \leq \mathcal{F}, \forall \{x_i, x_j\} \in E$
    Z dokázaného Lemma 3.17 vyplýva, že mi túto vlastnosť stačí ukázať pre hrany z $I(\mathcal{F})$. Intuitívne je to jednoduché: začnem s hranou $\{x_i, x_j\}$ a budem pokračovať vždy po hlavnej ceste (vrcholoch s označením H) a raz dôjdem k listu, ktorý mi

pridá premennú v. Ak by som chcel byť precízny, tak to môžem ukázať napríklad rozborom prípadov podľa toho, kde takýto term v algoritme pridávam.

- ∗ V kroku 2)b.
  Budem postupovať pomocou procedúry forward chaining. Začnem teda s množinou $\{p, x\}$ a budem sa chcieť dopracovať k tomu, aby som pridal v. Ako vyzerá postup procedúry forward chaining? Hneď v tomto kroku 2)b algoritmu k výslednej množine pridám vrchol x_1. Následne označím vložím do premenných p a x dve z prvkov výslednej množiny a pokračujem krokom 2). Pozriem sa na to, čo sa deje s p a x tentokrát. V princípe sú 2 možnosti. Buď sa k nim v kroku 2)a pridá v a som teda hotový (použil som len premenné z výslednej množiny). Alebo sa k nim pridá v kroku 2)b alebo 2)c ďalšia premenná x_1. Táto sa ale pridala pomocou termu, ktorý obsahuje ako pozitívne premenné len premenné z výslednej množiny, takže túto premennú môžme pridať k výslednej množine procedúry forward chaining. V oboch týchto prípadoch však algoritmus pokračuje opäť bodom 2). A tam bude buď pokračovať donekonečna opäť do bodov 2)b, 2)c, alebo raz dôjde do bodu 2)a a budeme hotoví. A keďže vieme, že algoritmus je konečný, určite nastane druhý z prípadov.
- ∗ V kroku 2)c.
  Dôkaz úplne rovnaký ako v predchádzajúcom prípade, iba s tým rozdielom, že tu sa hneď na začiatku dajú do výslednej množiny nie tri, ale hneď štyri premenné - $\{p, x, x_1, x_2\}$. Potom sa do premenných x a p vložia tiež dve prvky z tejto množiny. Ďalej analogicky.

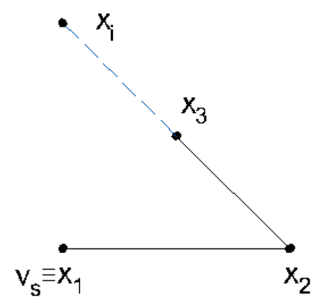
3.3.3 DNF → HV-pokrytie

Nech teda f_G má DNF reprezentáciu $\mathcal{F}_0$ s maximálne $m + 2$ termami. Pomocou modifikácií 0, 1, 2, 3 (z Lemma 3.3) vytvoríme DNF $\mathcal{F}$ splňujúcu všetky dôsledky tvrdení 3.3 až 3.9.
Na základe DNF $\mathcal{F}$ najprv určím vrchol x_S. Potom z tejto DNF budem postupne vytvárať K - kostru grafu G ako aj funkciu $f : V(K) \to \{H, V\}$. A to tak, že vždy vezmem jeden term z $\mathcal{F}$, spracujem ho a na základe toho buď obohatím množinu K, rozšírim funkciu f, alebo oboje či dokonca ani jedno. Každopádne budú f, K, x_S v každom kroku splňovať definíciu HV-pokrytia grafu $G' = (V, E)$, kde $V(G') = K$ a $x, y \in E(G')$ ak $x \in K, y \in K, x, y \in E(G)$. Inak povedané, G' odpovedá reštrikcii grafu G na vrcholy K. Navyše, až spracujem aj posledný term z $\mathcal{F}$, G' bude totožný z G. Tým vytvorím HV-pokrytie grafu G. Na začiatku si teda položím $V(K) = \emptyset, E(K) = \emptyset$, f nedefinovaná v žiadnom x_i, v_S nedefinovaný, fronty P a O prázdne. V ďalšom texte budem symbolom G' označovať graf G zúžený na vrcholy z aktuálnej množiny K, tak ako je to definované vyššie.

Takže začneme s množinou S.
Vieme, že existuje hrana $e_S \in E(G)$ taká, že $S = e_S$. Označme si BÚNO vrcholy tejto hrany ako $e_S = \{x_1, x_2\}$. Zároveň platí, že $e_S \in I, H_{e_S} \neq \emptyset$, takže existuje v $\mathcal{F}$ term $x_1 x_2 \bar{x}_i$ pre nejaké i, BÚNO nech je to x_3. Zároveň vieme, že $N_{e_S} \neq \emptyset$, takže v $E(G)$ existuje hrana $e_1 = \{x_1, x_3\}$ alebo $e_2 = \{x_2, x_3\}$ alebo oboje súčasne. Ak existuje hrana e_1, tak potom $\mathcal{F}$ obsahuje term $e_1 \bar{v}$ alebo $e_1 \bar{x}_i$ pre nejaké i. Teda $e_1 \in I(\mathcal{F})$ alebo $e_1 \in T(\mathcal{F})$. Obdobne to platí aj pre e_2. Predpokladajme, že $e_1 \in I(\mathcal{F}) \vee e_2 \in I(\mathcal{F})$.
Ak by totiž nastal prípad, že aj $e_1 \in T(\mathcal{F})$, aj $e_2 \in T(\mathcal{F})$, tak to by znamenalo, že G obsahuje

len tieto tri vrcholy x_1, x_2, x_3. Pretože ak by existoval nejaký ďalší, tak by sa k nemu DNF $\mathcal{F}$ pomocou procedúry forward chaining nikdy nemohla dostať z premennej v. Začala by s e_S, pridala x_3 a potom by už žiaden ďalší vrchol nemohla pridať. V takomto prípade položím $E(K) = \{\{x_1, x_2\}, \{x_2, x_3\}\}, f(x) = H \forall x \in V(G), v_S = x_1$. Nie je ťažké si rozmyslieť, že to odpovedá HV-pokrytiu grafu G. V nasledújúcom texte teda predpokladajme, že G je aspoň 4-prvokvý.
Mňa teda zaujímajú len situácie, kedy sú tieto hrany v $I(\mathcal{F})$. Rozdelím si dva prípady:

a) Práve jeden z e_1, e_2 je v $I(\mathcal{F})$ - obrázok 3.8

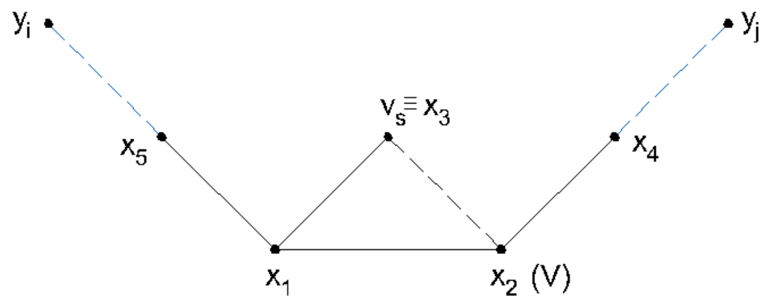


Obr. 3.8: Začiatok cesty - $\{x_2, x_3\} \in I(\mathcal{F})$.

(druhá hrana je buď v $T(\mathcal{F})$, alebo neexistuje vôbec). BÚNO predpokladajme, že je to e_2. Máme teda termy $e_1\bar{x_3}, e_2\bar{x_i}$ pre nejaké i. Položme $V(K) = V(K) \cup \{x_1, x_2\}, E(K) = E(K) \cup \{x_1, x_2\}, v_S = x_1, f(x_1) = f(x_2) = H$. Nie je ťažké nahliadnuť, že K, f, v_S splňujú podmienky HV-pokrytie pre graf G'. Napríklad to, že $f(x_1)$ bude vždy H plynie z defínicie HV-pokrytia. Tak aj to, že $f(x_2) = H$ nikdy neporuší podmienky defínície, je odvoditeľné. Buď nebude susediť so žiadnym vrcholom stupňa 3 - vtedy je jeho označenie ako H nespochybniteľné. Rovnako tak aj v prípade, že z neho pôjde ešte ďalšia hrana a stane sa teda vrcholom stupňa 3 (napríklad tým, že v $\mathcal{F}$ bude term $x_2x_i\bar{x_j}$ pre nejaké j). Zostáva posledný prípad a síce že bude susediť s nejakým vrcholom stupňa 3. Vtedy ale bude nutne najbližšie k x_1, pretože z x_1 už nevedie žiadna ďalšia hrana.
$P := P \cup (x_3, x_2x_3\bar{x_i})$
Ešte poznámka k obrázkom: plné čierne hrany buď sú v K, alebo tam s istotou budú patriť. Modrá čiarkovaná čiara znázorňuje "smer rozširovania" - tj. ak je medzi dvoma premennými modrá čiara, v nasledujúcom kroku sa spracuje term, ktorý obsahuje ako pozitívne literály oba jej konce.

b) Obe hrany sú v $I(\mathcal{F})$, teda $e_1 \in I(\mathcal{F}), e_2 \in I(\mathcal{F})$.
Máme teda termy $x_1x_2\bar{x_3}, x_1x_3\bar{x_i}, x_2x_3\bar{x_j}$ pre nejaké i, j. BÚNO nech $i = 5, j = 4$. Takže graf G obsahuje cyklus $x_1x_2x_3$. Zároveň z toho plynie, že graf G obsahuje buď hranu $\{x_1, x_5\}$ alebo hranu $\{x_3, x_5\}$ (alebo oboje súčasne). To isté platí pre hrany $\{x_2, x_4\}, \{x_3, x_4\}$. Teraz sa pozrime čo sa deje s vrcholom x_3. Už vieme, že bol použitý v dvoch hranách. Teda sa môže vyskytnúť už len v jednej alebo v žiadnej. Takže máme opäť dva prípady:

(a) x_3 sa už nevyskytne v ďalšej hrane - obrázok 3.9



Obr. 3.9: Začiatok cesty - $e_1, e_2 \in I(\mathcal{F}), x_3$ už nebude v ďalšej hrane.

Takže nám pribudnú termy $x_1x_5\bar{y}_i, x_2x_4\bar{y}_j$, kde y_i, y_j sú buď ďalšie premenné, alebo v. Teraz položím: $v_S = x_3, V(K) = V(K) \cup \{x_1, x_2, x_3\}, E(K) = E(K) \cup \{\{x_1, x_3\}, \{x_1, x_2\}\}, f(x_3) = f(x_1) = H, f(x_2) = V$. Opäť môžme overiť, že K je HV-pokrytím G'. Vieme, že x_1 bude v K stupňa 3, má označenie H. x_2 bude mať stupeň 2 (jednu hranu obsahujúcu x_2 som už "vyhodil hranu $\{x_2, x_3\}$), je označený ako V, dostanem sa do neho po hrane $\{x_2, x_3\} \in G' \setminus K$ z vrcholu $x_3 = v_S$. Rozhodnem sa čo s termom $x_2x_4\bar{y}_j$:

- $y_j = v$
  $V(K) := V(K) \cup \{x_4\}, E(K) := E(K) \cup \{x_2, x_4\}, f(x_4) = H$
  Týmto som neporušil žiadnu podmienku HV-pokrytia.
- $y_j = x_j \in V(G)$
  $P := P \cup (x_4, x_2x_4\bar{x}_j)$

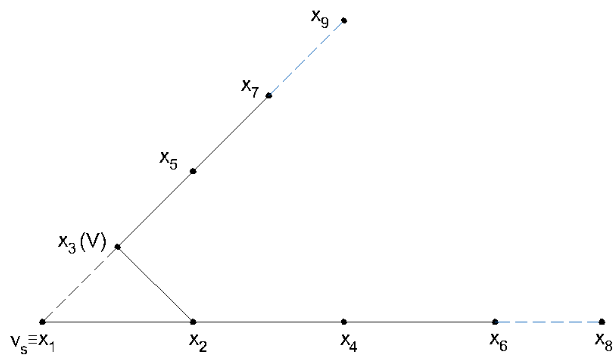
Rozhodnem sa čo s termom $x_1x_5\bar{y}_i$:

- $y_i = v$
  $V(K) := V(K) \cup \{x_5\}, E(K) := E(K) \cup \{x_1, x_5\}, f(x_5) = H$
  Týmto som opäť neporušil žiadnu podmienku HV-pokrytia.
- $y_j = x_6 \in V(G)$
  Keďže x_1 je v G stupňa 3, k termu $x_1x_5\bar{x}_6$ musí existovať term $x_5x_6\bar{y}_7$. Podľa toho ako vyzerá sa rozhodnem:
  - $y_7 = v$
    $V(K) := V(K) \cup \{x_6\}, E(K) := E(K) \cup \{x_5, x_6\}, f(x_6) := H$
  - $y_7 = x_7 \in V(G)$
    $P := P \cup (x_6, x_5x_6\bar{x}_7)$

  Ani v jednom z vyššie uvedených prípadov som neporušil podmienky HV-pokrytia či podmienky kladené na P (ako sa dozvieme neskôr).

(b) x_3 sa ešte vyskytne v jednej z hrán,
BÚNO nech v hrane odpovedajúcej termu $x_3x_5\bar{x}_7$. Potom nutne existuje aj term $x_2x_4\bar{x}_6$. Ak uvážime všetky možné termy, ktoré môžu vzniknúť z vyššie uvedených $x_1x_2\bar{x}_3, x_1x_3\bar{x}_5, x_2x_3\bar{x}_4$, tak môžu vzniknúť už len posledné dve možnosti:

i. Nevznikol už žiaden ďalší term - obrázok 3.10.



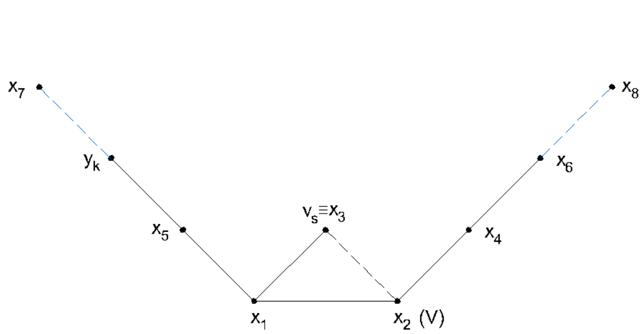
Obr. 3.10: Začiatok cesty - $e_1, e_2 \in I(\mathcal{F}), x_3$ v hrane, žiaden ďalší term.

Tj. máme termy $x_1x_2\bar{x_3}, x_1x_3\bar{x_5}, x_2x_3\bar{x_4}$ a $x_3x_5\bar{x_7}, x_2x_4\bar{x_6}$. Teraz môžem položiť: $v_S = x_1, V(K) = V(K)\cup\{x_1, x_2, x_3, x_4, x_5\}, E(K) = E(K)\cup\{\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_5\}, \{$ $f(x_2) = f(x_4) = f(x_5) = H, f(x_3) = V$. Opäť sa môžeme jednoducho presvedčiť, že K teraz splňuje všetky podmienky HV-pokrytia grafu G'.
Termy $x_2x_4\bar{x_6}$ a $x_3x_5\bar{x_7}$ sú v $I(\mathcal{F})$. Takže z nich určite vzniknú ďalšie hrany. Z prvého vznikne $x_4x_6\bar{x_8}$ (lebo inak by bol vrchol x_2 stupňa 4) a z druhého vznikne $x_5x_7\bar{x_9}$ (lebo vrchol x_3 by bol stupňa 4).
$P := P \cup \{(x_6, x_4x_6\bar{x_8}), (x_7, x_5x_7\bar{x_9})\}$

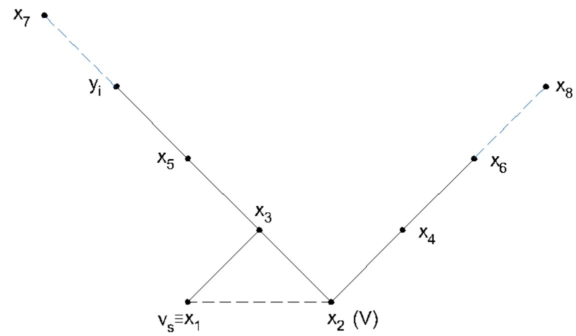
ii. Vznikol ešte posledný možný term, a to $x_1x_5\bar{y_k}$ - obrázok 3.11.

Takže máme termy $x_1x_2\bar{x_3}, x_1x_3\bar{x_5}, x_2x_3\bar{x_4}$ a $x_3x_5\bar{y_i}, x_1x_5\bar{y_k}, x_2x_4\bar{x_6}$. Schválne som v tomto výčte nechal namiesto nových premenných iba y_i, y_k. A to z toho dôvodu, že jedna z týchto premenných musí byť nutne rovná v.
Ak by totiž obidve tieto premenné odpovedali premenným nejakého vrcholu, tak by sme dostali nové termy $x_3x_5\bar{x_7}, x_1x_5\bar{x_8}$. Oba tieto termy sú prvkami $I(\mathcal{F})$. Teda N_e je neprázdne pre oba. V prípade prvého termu by teda v grafe G musela existovať hrana $\{x_3, x_7\}$ alebo $\{x_5, x_7\}$. Prvá možnosť nemôže nastať, lebo x_3 by malo stupeň 4. Takže existuje hrana $\{x_5, x_7\}$. Teraz ale nemôže existovať ani hrana $\{x_1, x_8\}$ (lebo x_1 by malo stupeň 4), ani hrana $\{x_5, x_8\}$ (lebo x_5 by malo stupeň 4).
Vieme teda, že jedno z y_i, y_j musí byť rovné premennej v. Robím to nerád, ale musím tu vložiť ešte jedno rozdelenie na dva prípady:

A. $y_i = v$
Položíme $v_S = x_3, V(K) = V(K) \cup \{x_1, x_2, x_3, x_4, x_5\}, E(K) = E(K) \cup \{\{x_1, x_2\}, \{x_1, x_3\}, \{x_1, x_5\}, \{x_2, x_4\}\}, f(x_1) = f(x_3) = f(x_4) = f(x_5) = H, f(x_2) = V$. Aj teraz je K HV-pokrytím grafu G'.
Uvážim ešte term $x_1x_5\bar{y_k} \in I(\mathcal{F})$. Z neho musí vzniknúť ďalšia hrana, $\{x_1, y_k\}$ to nemôže byť (x_1 by malo stupeň 4), takže to musí byť $\{x_5, y_k\}$. Nech sa tak stane pomocou termu $x_5y_k\bar{x_7}$. Z podobného dôvodu musí k termu $x_2x_4\bar{x_6}$ existovať term $x_4x_6\bar{x_8}$.
$P := P \cup \{(y_k, x_5y_k\bar{x_7}), (x_6, x_4x_6\bar{x_8})\}$



(a) $y_i = v$



(b) $y_k = v$

Obr. 3.11: Začiatok cesty - $e_1, e_2 \in I(\mathcal{F}), x_3$ v hrane, ešte ďalší term.

B. $y_k = v$
Položíme $v_S = x_1, V(K) = V(K) \cup \{x_1, x_2, x_3, x_4, x_5\}, E(K) = E(K) \cup \{\{x_1, x_3\}, \{x_2, x_3\}, \{x_3, x_5\}, \{x_2, x_4\}\}, f(x_1) = f(x_3) = f(x_4) = f(x_5) = H, f(x_2) = V$. Aj teraz je K HV-pokrytím grafu G'.
Uvážim ešte term $x_3 x_5 \bar{y_i} \in I(\mathcal{F})$. Z neho musí vzniknúť ďalšia hrana, $\{x_3, y_i\}$ to nemôže byť (x_3 by malo stupeň 4), takže to musí byť $\{x_5, y_i\}$. Nech sa tak stane pomocou termu $x_5 y_i \bar{x_7}$. Z podobného dôvodu musí k termu $x_2 x_4 \bar{x_6}$ existovať term $x_4 x_6 \bar{x_8}$.
$P := P \cup \{(y_i, x_5 y_i \bar{x_7}), (x_6, x_4 x_6 \bar{x_8})\}$

Dovolím si ešte poznámku. Vyššie v jednotlivých možnostiach som vymenuvával premenné až do x_9. Ak by sa kedykoľvek stalo, že by ten graf bol menší, to jest že by namiesto tej premennej y_i v tom dôkaze mala vystupovať premenná v (nech je to v terme $x_1 x_2 \bar{y_i}$), tak by som sa zachoval podobne ako v prípade b)a. To znamená, že by som takýto term nevložil do P, ale by som ho rovno spracoval. A to v tom zmysle, že by som do K vložil vrchol x_2, hranu $\{x_1, x_2\}$, a vrchol x_2 označil ako H.

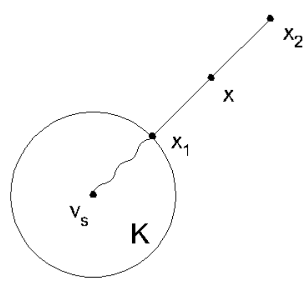
Indukčný predpoklad.
V doterajšom priebehu som si teda založil graf K, vyznačil v_S, pre prvky K nadefinoval hodnoty funkcie f. A to tak, že dohromady splňujú podmienky HV-pokrytia grafu G'. Inak povedané, dokázal som indukčný predpoklad. Teraz dokážem krok od grafu K ku grafu $K \cup x_i$ pre nejaké i tak, aby sa zachovali vlastnosti HV-pokrytia (to všetko samozrejme s pomocou našej DNF $\mathcal{F}$).
Podmienky kladené na v_S boli vo všetkých prípadoch splnené, túto premennú už meniť nebudem. To znamená, že v ďalšom texte už nemusím zakaždým odvodzovať, prečo príslušná v_S splňuje všetky podmienky HV-pokrytia na ňu kladené.

Fronta P.
V každom kroku som si do fronty P vložil dvojicu (x, T), kde x je premenná a $T \in \hat{\mathcal{F}}$ je term tvaru $x_1 x \bar{x_2}$, kde x_1, x_2 sú nejaké vrcholy grafu G (ich pomenovanie číslicami 1 a 2 len pre zjednodušenie). Pričom platí, že $x_1 \in K$, a teda existuje HV-cesta z v_S do v_1. Zároveň $x \notin K, x_2 \notin K$. Z toho vyplýva, že x_1 má k vrcholu v_S najmenšiu vzdialenosť (v K) spomedzi všetkých troch (z vrcholu v_S sa k x, x_2 môžem dostať jedine cez x_1). Ďalej platí, že x_1 je v K buď stupňa 1, alebo je stupňa 2 a vedie z neho v K hrana do nejakého vrcholu x_i takého, že

$f(x_i) = V$. Všetky tieto podmienky budú aj v naďalej vždy splnené, keď budem do fronty P niečo vkladať.
V nasledujúcom texte keď budem písať, že medzi nespracovanými termami z $\mathcal{F}$ (alebo v O) hľadám pre x taký term, ktorý je tvaru $x_1 x \bar{x}_2$, budem tým prirodzene myslieť, že mi vyhovuje aj term $x x_1 \bar{x}_2$. Do fronty P však tieto termy uložím vždy tak, aby splňovali vyššie uvedené vlastnosti. To znamená, že ak bude potrebné, BÚNO prehodím nenegované premenné tak, aby tieto podmienky splňovali.

Fronta O.
Táto fronta bude obsahovať "odbočky" z hlavnej cesty. Pre jeden prvok fronty $(x, T) \in O$ bude platiť, že T je tvaru $x_1 x \bar{x}_2$. Kde $x_1 \in K$. Spomeniem ešte, že v priebehu algoritmu pridám do K všetky x z O a to tak, že budu hranou (z K) spojené s x_1. (Samozrejme ak budú stupňa väčšieho ako 1, tak aj hranou z $E(G \setminus K)$ s nejakým vrcholom z K.)
Ak v ďalšom texte pre nejaký vrchol poviem"$x_i \in O$", bude to znamenať, že v O existuje nejaký prvok tvaru (x_i, S), S ľubovoľné (prirodzene až na vyššie uvedené pravidlá). Podobne ak poviem "$R \in O$" pre nejaký term R, bude to znamenať, že v O existuje prvok tvaru (x_j, R) pre nejaké x_j.

Teraz sme pripravení, aby sme mohli pristúpiť k popisu jedného kroku algoritmu, ktorý z DNF $\mathcal{F}$ vytvára HV-pokrytie. Takýto krok je znázornený na obrázku 3.12, kde $x_1 \in K$.



Obr. 3.12: Jeden krok algoritmu. Práve spracovávaným termom je $x_1 x \bar{x}_2$.

1) Vyber prvý prvok z fronty P,
vlož ho do (x, T). Ak bola fronta P prázdna a nič si nevybral, choď na krok 6). Inak choď na krok 2).

2) Zisti, ako vyzerá pokračovanie cesty.
$T = x_1 x \bar{x}_2$, medzi nespracovanými termami teda musí existovať term tvaru $x_1 x_2 \bar{y}_3$ alebo term tvaru $x x_2 \bar{y}_4$ alebo oboje súčasne. Na základe toho sa rozhodnem:

   (a) Ak existujú oba a $y_3 \neq v, y_4 \neq v$, choď na krok 5).

   (b) Ak existujú oba a $y_4 \neq v, y_3 = v$, alebo ak existuje len $x x_2 \bar{y}_4$ (nech je y_4 čokoľvek), choď na krok 3).

   (c) Ak existujú oba a $y_3 \neq v, y_4 = v$, alebo ak existuje len $x_1 x_2 \bar{y}_3$(nech je y_3 čokoľvek), choď na krok 4).

(d) Zostal už len prípad, kedy existujú oba a $y_3 = v, y_4 = v$. Teraz si môžem vybrať, či budem pokračovať krokom 3) alebo krokom 4). BÚNO pokračuj krokom 3).

V každom prípade označ oba termy v $\mathcal{F}$ ako spracované.

3) Pôjdem rovno.
$V(K) := V(K) \cup \{x\}, E(K) := E(K) \cup \{x_1, x\}, f(x) := H$

- $y_4 = v$
  $V(K) := V(K) \cup \{x_2\}, E(K) := E(K) \cup \{x, x_2\}, f(x_2) := V$
- $y_4 = x_4$, pre nejaké $x_4 \in V(G \setminus K)$
  $P := P \cup (x_2, xx_2\bar{x_4})$

Pokračuj krokom 1).

4) Nasleduje odbočka.
$V(K) := V(K) \cup \{x_2\}, E(K) := E(K) \cup \{x_1, x_2\}, f(x_2) := H, O := O + (x, x_1x\bar{x_2})$

- Ak $y_3 \neq v$, tj. $y_3 = x_3, x_3 \in V(G \setminus K)$, potom je term $x_2x_3\bar{y_5}$ prvkom množiny nespracovaných termov z $\mathcal{F}$ (kde $y_5 = v$ alebo $y_5 = x_5, x_5 \in V(G \setminus K)$). Označ tento term ako spracovaný a:
  - $y_5 = v$
    $V(K) := V(K) \cup \{x_3\}, E(K) := E(K) \cup \{x_2, x_3\}, f(x_3) = H$
  - $y_5 = x_5$
    $P := P \cup (x_3, x_2x_3\bar{x_5})$

Pokračuj krokom 1).

5) Cesta sa rozdvojuje.
$V(K) := V(K) \cup \{x, x_2\}, E(K) := E(K) \cup \{\{x_1, x\}, \{x_1, x_2\}\}, f(x) := H, f(x_2) := V$.
Existuje nespracovaný term $x_2x_3\bar{y_5}$. Označ ho ako spracovaný a:

- $y_5 = v$
  $V(K) := V(K) \cup \{x_3\}, E(K) := E(K) \cup \{x_2, x_3\}, f(x_3) := H$
- $y_5 = x_5$
  $P := P \cup (x_3, x_2x_3\bar{x_5})$

Zároveň existuje aj nespracovaný term $xx_4\bar{y_6}$. Označ ho ako spracovaný a:

- $y_6 = v$
  $V(K) := V(K) \cup \{x_4\}, E(K) := E(K) \cup \{x, x_4\}, f(x_4) := H$
- $y_6 = x_6$
  $P := P \cup (x_4, xx_4\bar{x_6})$

Pokračuj krokom 1).

6) Nájdenie napojujúcej hrany.
   Spomedzi nespracovaných termov z $\mathcal{F}$ vyber jeden tvaru $S = x_1 x \bar{x_2}$, kde obe pozitívne premenné sú buď v O alebo v K a negatívna premenná nie je z K. Teda $(x_1 \in K \vee x_1 \in O) \wedge (x \in K \vee x \in O) \wedge x_2 \in V(G \setminus K)$.
   Ak žiaden takýto term neexistuje, pokračuj krokom 9).
   Ak existuje, označ tento term ako spracovaný a dohľadaj termy $x_1 x_2 \bar{y_3}, x x_2 \bar{y_4}$. Maximálne jeden z y bude premenná, druhý bude v (inak by x_2 malo stupeň 4). Nech je to y_4 (x je teda vrchol, ktorý sa napojí na x_2). Môže nastať aj prípad, kedy existuje len jeden z týchto termov, vtedy tento term označím $x_s x_2 \bar{y_5}$.
   Ak (existujú oba a $x_1 \in O$) alebo ($y_3 = v, y_4 = v$ a jedna z x, x_1 je prvkom O - nech je to x_1) :

   - Z O vyber záznam patriaci k x_1, nech je to $T = x_0 x_1 \bar{x_h}$.
     $V(K) := V(K) \cup \{x_1\}, E(K) := E(K) \cup \{x_0, x_1\}, f(x_1) := V$
     (Týmto zaručím, že v ďalších krokoch bude skutočne prvá premenná v terme v K a teda prípadné vloženie tohto termu do P neporuší podmienky.)

   A ďalej ak:

   - existujú oba a $x \in K$ alebo existuje len jeden a $x_s \in K$ (nezávisle na y_5).
     Pokračuj krokom 7).
   - existujú oba a $x \in O$ alebo existuje len jeden a $x_s \in O$ (nezávisle na y_5).
     Pokračuj krokom 8).
   - existujú oba a $y_3 = v, y_4 = v$.
     V súlade s vyššie uvedeným, ak jedno z x, x_1 bolo prvkom O, tak to bolo x_1 a už je vložené v K.
     Teraz sa rozhodneme podľa druhého
     - $x \in K$
       Pokračuj krokom 7).
     - $x \in O$
       Pokračuj krokom 8).

7) Napojenie na koniec.
   Máme teda term $S = x_1 x \bar{x_2}, x_1 \in K, x \in K$. Aspoň jeden z x_1, x musí byť stupňa 1 v K. BÚNO nech je to x_1. Existuje teda term $x_1 x_2 \bar{y_3}$.

   - $y_3 = v$
     $V(K) := V(K) \cup \{x_2\}, E(K) := E(K) \cup \{x_1, x_2\}, f(x_2) := H$
     Pokračuj krokom 6).
   - $y_3 = x_3$
     $P := P \cup (x_2, x_1 x_2 \bar{x_3})$
     Pokračuj krokom 1).

8) Napojenie odbočky.
   V kroku 6) sme našli napojujúci term $S = x_1 x \bar{x_2}, x_1 \in K$. Vieme, že $x \in O$, teda v O existuje term tvaru $T = x_0 x \bar{x_h}$. Vyber ho z O a:
   $V(K) := V(K) \cup \{x\}, E(K) := E(K) \cup \{x_0, x\}, f(x) := V$

Medzi nespracovanými termami $\mathcal{F}$ existuje term tvaru $xx_2\bar{y_3}$. Označ ho ako spracovaný a:

- $y_3 = v$
  $V(K) := V(K) \cup \{x_2\}, E(K) := E(K) \cup \{x, x_2\}, f(x_2) := H$ Pokračuj krokom 6).
- $y_3 = x_3$
  $P := P \cup (x_2, xx_2\bar{x_3})$
  Pokračuj krokom 1).

9) Napojenie odbočiek dĺžky 1.
Pre všetky $(x, T) \in O, T = x_1x\bar{x_2}$:

- $V(K) := V(K) \cup \{x\}, E(K) := E(K) \cup \{x_1, x\}, f(x) := V$

Overenie kroku 2) algoritmu.
Pri rozhodovaní o ďalšom toku algoritmu v tomto bode sa stalo, že viaceré možnosti z tých, ktoré môžu nastať, som poslal do rovnakého kroku algoritmu. Chce len málo námahy overiť, že všetky tieto kroky sú skutočne korektné pre každú z týchto možností.

Overenie kroku 3) algoritmu.
Mám spracovať term tvaru $x_1x\bar{x_2}$, pričom viem, že existuje aj term $xx_2\bar{y_4}$ (a pre žiadne y_i neexistuje term tvaru $x_1x_2\bar{y_i}$). Viem, že $x_1 \in K, deg_K(x_1) = 1$. Takže ak pridám x do grafu K, neporuším okamžite podmienku e) definície HV-pokrytia (ak budem chcieť neskôr pridať do K nejaký vrchol x_i tak, aby bol spojený hranou s x_1, tak ho označím ako V, čiže položím $f(x_i) = V$). Podmienky d) a c) už neskôr v procese porušiť vrcholom x nemôžem, keďže som ho označil ako H. Zostáva podmienka f). Z v_S vedie HV-cesta do vrcholu x_1 (indukčný predpoklad), do K som pridal hranu $\{x_1, x\}$.
Prípad $y_4 = v$. Pre vrchol x tu platia rovnaké argumenty ako vyššie pre x_1, takže x_2 neporušuje podmienku e). Rovnako ani podmienky d), c). Podmienka f) je splnená tým, že, ako som vyššie ukázal, do vrcholu x vedie HV-cesta, pridal som aj hranu $\{x, x_2\}$.
Overím ešte obsah P. $x \in K$, vyššie som ukázal, že do neho ide HV-cesta z v_S. x_2, x_4 nie sú z K. x je stupňa 1 (v K je spojený len s x_1).

Overenie kroku 4) algoritmu
Mám spracovať term tvaru $x_1x\bar{x_2}$, pričom viem, že existuje aj term $x_1x_2\bar{y_3}$ (a pre žiadne y_i neexistuje term tvaru $xx_2\bar{y_i}$).
Pridal som vrchol x_2, hranu $\{x_1, x_2\}$, existuje HV-cesta z v_S do x_1, platí teda podmienka f) definície HV-pokrytia. Podobne platia a platiť vždy budú podmienky d), c). Neskôr v algoritme pridám z fronty O do grafu K aj vrchol x a to s označením V. To bude ale v poriadku, pretože ho hranou pripojím na vrchol x_1, ktorý je H a je spojený hranou s x_2, ktorý je tiež H. Podmienka e) bude teda splená pre vrchol x_1.
Je mojou povinnosťou ešte objasniť, prečo v prípade $y_3 \neq v$, nevyhnutne existuje $x_2x_3\bar{y_5}$ pre nejaké y_5. Tak to je jednoduché. Keďže vieme, že existuje $x_1x_2\bar{x_3}$, tak musí existovať aj hrana $\{x_1, x_3\}$ alebo $\{x_2, x_3\}$. Ale x_1 už má stupeň 3, musí to byť druhá z menovaných. A teda musí term existovať (a to buď $y_5 = v$, alebo $y_5 = x_5, x_5 \in V(G \setminus K)$).
Prípad $y_5 = v$: x_2 je v K, existuje teda HV-cesta k v_S, vložil som aj hranu $\{x_2, x_3\}$, existuje teda HV-cesta od v_S aj pre x_3. Aj ostatné podmienky definície HV-pokrytia sú splnené.

Prípad $y_5 = x_5$: $x_2 \in K$, $x_3 \notin K, x_5 \notin K$, $\deg_K(x_2) = 1$.

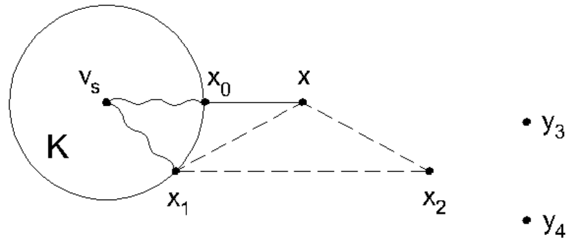
Overenie kroku 5) algoritmu.
Vrchol x_1 je stupňa 3, má označenie H. Jeho susedia majú jeden H, druhý V. Vrchol x je spojený hranou s x_1, má teda HV-cestu. Vrchol x_2 je V, je susedom vrcholu stupňa 3. Zároveň je hranou (z K) spojený s x_1. Ale čo je dôležitejšie, je hranou $\{x, x_2\} \in E(G \setminus K)$ spojený s vrcholom x, $f(x) = H$, existuje HV-cesta z v_S do x. Poznamenám len, že hrana $\{x, x_2\}$ sa do K ani nikdy nepridá, pretože okrem termu $x_2 x_3 \bar{y}_5$ sa už nebude spracovávať žiaden term obsahujúci x_2 (lebo inak by bol x_2 stupňa väčšieho ako 3). A z tohto termu sa hrana $\{x_2, y_5\}$ určite nepridá, lebo takisto by bolo x_2 stupňa väčšieho ako 3.
Zaujímavosť: tento krok algoritmu som nazval "rozdvojka", pretože v ňom vrcholy x a x_2 vystupujú ako úplne rovnocenné. Sú vzájomne prepojené hranou (mimo K). Takže som sa mohol ľubovoľne rozhodnúť, ktorý z nich bude Hlavný a ktorý Vedľajší.

Overenie kroku 6) algoritmu.
Nemôže sa stať, že sa premenná x vyskytne v terme typu $x_1 x \bar{x}_2$ tak, aby $x_2 \in K$. To by porušovalo tú vlastnosť modifikovanej minimálnej DNF, že $\forall e \neq e' : H_e(\mathcal{F}) \cap H_{e'}(\mathcal{F}) = \emptyset$.
V prípade $x_1 \in O$ som si mohol dovoliť vložiť do K aj premennú x. Zároveň viem, že k nej príslušný záznam v O už nebudem nikdy potrebovať. Ak by sa totiž aj stalo, že by ešte existoval nejaký term typu $x_i x_1 \bar{x}_j$, tak by bola vynútená existencia termu $x_i x_j \bar{x}_k$ (inak by bolo x_1 stupňa 4). Z toho plynie, že bude nutne $\deg_K x_1 = 1$ a teda je zachovanie HV-cesty k tomuto vrcholu zaručené napojením na x_0.
To, že v tomto kroku algoritmus vyberie skutočne všetky zo zvyšných termov nachádzajúcich sa medzi ešte nespracovanými, je podrobne objasnené v časti Úplnosť algoritmu.

Overenie kroku 7) algoritmu.
Jeden z x_1, x musí byť stupňa 1 preto, lebo ak by boli obaja stupňa 2, tak hranou $\{x_1, x\}$ by sa stali stupňom 3. Ale kvôli termu $x_1 x \bar{x}_2$ by jeden z nich musel byť napojený na x_2. Spor.
V prípade, že boli obe premenné x_1, x stupňa 1, tak mohli existovať oba termy $x_1 x_2 \bar{y}_5, x x_2 \bar{y}_6$. V tomto prípade by však aspoň jeden z nich musel byť rovný v (inak by bol vrchol x_2 stupňa 4). BÚNO nech by to bol y_5. V tomto prípade nech sa algoritmus zachová úplne rovnako ako je v ňom popísané. Nič sa nepokazí.
Keďže $x_1 \in K$, máme splnený predpoklad existencie HV-cesty a vložením x_2 sa predĺži. Aj podmienky kladené na frontu P sa zachovajú.

Overenie kroku 8) algoritmu.
Tento krok je schématicky znázornený na obrázku 3.13. Vložili sme x, prítomnosť HV-cesty je zaručná napojením na $x_1 \in K$.

Tvrdím tam, že existuje term tvaru $x x_2 \bar{y}_3$. To vyplýva z toho, že jediný spôsob ako sa sem dostať je z kroku 6) algoritmu. No a tam prepošlem tok do tohto bodu len v prípade, že takýto term existuje.
V tomto kroku sa nestarám o prípadný term $x_1 x_2 \bar{y}_4$. Je to tak preto, lebo z kroku 6) sem prídem jedine v stave, kedy by takýto term neexistoval, alebo by prípadné y_4 bolo rovné v. Ani jeden z týchto prípadov nemusím riešiť, lebo neporušuje žiadnu z vlastností HV-pokrytia či vlastností kladených na P, O.



Obr. 3.13: Napojenie odbočky. $x_0, x_1 \in K$.

Overenie kroku 9) algoritmu.
Keďže som v kroku 6) už nevybral žiaden term spomedzi nespracovaných, tak žiaden nespracovaný term neexistuje (toto podrobnejšie odvodím nižšie v časti Úplnosť algoritmu). Zároveň je fronta P prázdna (inak by som sa sem nedostal). Takže všetky spracované premenné sú buď v K alebo v O (objasnené v časti Úplnosť algoritmu). Pod spracovanými termami mám na mysli premenné, ktoré vystupovali v nejakom terme z $I(\mathcal{F})$ ako negatívne. Vieme, že každá premenná je v nejakom z týchto termov. Z toho dostávame, že každý vrchol je teraz buď v K alebo v O. Žiadna premenná z O nie je zároveň aj v K. Takže môžme ku K pripojiť všetky prvky z O, pretože budú stupňa 1 v K a teda sa na nich nebude klásť požiadavka, aby boli hranou z $V(G \setminus K)$ spojené s nejakým vrcholom z K.
A stupňa 1 budú práve preto, že všetky premenné sú už spracované (sú v K). A tie, ktoré sú v O budú spojené hranou priamo ku K, takže pomoc tohto vrchola nebudú potrebovať.

Konečnosť algoritmu.
V každom cykle algoritmu sa označí aspoň jeden nespracovaný term tvaru $x_1x_2\bar{x}_3$ z $\mathcal{F}$ ako spracovaný (za cyklus algoritmu môžme považovať napríklad každú situáciu, kedy sa vráti do bodu 1) alebo 6)). V súlade s predchádzajúcim značením, x_3 označuje premennú odpovedajúcu vrcholu grafu, teda $x_3 \neq v$. Týchto termov je konečný počet (presne n - počet vrcholov grafu). Teda po n krokoch nebude mať algoritmus čo označovať a bude pokračovať krokom 9). Krok 9) skončí v čase $O(n)$.

Zložitosť algoritmu.
Všetky kroky algoritmu (okrem 6) a 9)) môžu pri vhodne zvolených datových štruktúrach trvať konštantný čas. Krok 6) potom trvá až $O(n)$. To nám pre kroky 1) až 8) dáva zložitosť $O(n^2)$. Krok 9) je $O(n)$. Inicializačný proces (nájdenie prvku v_S a počiatočné naplnenie fronty P) môžme zvládnuť v konštantnom čase. Vytvorenie vhodných datových štruktúr v $O(m)$. Takže celý algoritmus dohromady dáva $O(n^2)$.

Korektnosť algoritmu.
Vyššie je overovanie korektnosti jednotlivých krokov algoritmu. To jest, že v každom kroku vytvorená množina K, funkcia f, vrchol v_S splňovali jednotlivé podmienky HV-pokrytia. Čo som však neoveroval je, že či mi novopridané vrcholy nevytvorili cyklus (a tým by porušili podmienku, že K je kostra grafu G). Pre inicializačný proces (graf indukčného predpokladu) to je veľmi ľahké overiť rozborom jednotlivých prípadov. Teraz overím jednotlivé kroky algoritmu. V krokoch 1), 2) som do K nič nepridával. V krokoch 3), 4), 5), 7) som vždy pridával len vrcholy x, ktoré sa nachádzali

- v terme $T = x_1 x \bar{x_2} \in P$. Ale vieme, že vlastnosť P, ktorá hovorí o tom, že $x_1 \in K, x \notin K, x_2 \notin K$, sme po celý čas zachovávali a jej zachovanie overovali v jednotlivých krokoch.

- v ešte nespracovanom terme $T = x_i x_j \bar{x}$. Ale keďže term nebol ešte spracovaný, nemohol som nikdy predtým pridať premennú x do K (inak by sa u $\mathcal{F}$ porušila podmienka $\forall e \neq e' : H_e(\mathcal{F}) \cap H_{e'}(\mathcal{F}) = \emptyset$.

V krokoch 6), 8) a 9) som (mimo už odôvodnených prípadov) pripájal len vrcholy x, ktoré sa v O nachádzali v terme tvaru $x_1 x \bar{x_2}$. Keď som v priebehu algoritmu do fronty O dal term tohoto tvaru, nikdy som zároveň toto x nepridal do množiny K. Zároveň som tento term označil ako spracovaný (takže som ho nemohol použiť znova). A nepridal som ho ani do fronty P. A v ostatných krokoch algoritmu, v ktorých sa pridávajú vrcholy do K, sa spracovávali len termy z P. A z tých termov sa do K vkladali len premenné, ktoré boli "v strede" (a tie sa potom zahodili, nedávali sa už do O).

Úplnosť algoritmu.

Je každý vrchol z G skutočne aj v K? Vieme, že v $\mathcal{F}$sa každý vrchol nachádza v nejakom terme ako negovaná premenná. Spracoval som každý term z $\mathcal{F}$, ktorý má ako negovanú premennú $x_i \neq v$. Vždy, keď som nejaký term označil za spracovaný, bolo zároveň zabezpečené, že sa premenná x_2, ktorá je v ňom ako negovaná, určite dostane do K. A to tak, že sa do K priamo vložila, alebo sa vložila do P ako súčasť termu tvaru $x_i x_2 \bar{x_j}$ (teda ako "stredná"). Keď si pozrieme čo sa v jednotlivých prípadoch deje s premennou, ktorá je v strede (v popise algoritmu označená ako x), tak sú opäť dve možnosti – buď sa priamo vloží do K, alebo sa vloží do O ako stredná. A keď už nie v kroku 8) či 6), tak v kroku 9) sa každá takáto premenná do K určite dostane.

Ešte poslednou otázkou zostáva: spracoval som z $\mathcal{F}$ skutočne každý term tvaru $x_1 x_2 \bar{x_3}$? V kroku 6) algoritmu vyberiem ľubovľný nespracovaný term z $\mathcal{F}$. Takže jediný spôsob ako ho v tomto kroku nevybrať je, že nesplnil požiadavky na neho v bode 6) kladené. A síce, že musí byť tvaru $S = x_1 x \bar{x_2}$, kde $(x_1 \in K \vee x_1 \in O) \wedge (x \in K \vee x \in O) \wedge x_2 \in V(G \setminus K)$. Takže pre spor predpokladajme, že existujú nespracované termy a žiaden z nich algoritmus v kroku 6) nevybral. Ktorá z podmienok mohla byť problémom?

- $x_2 \in V(G \setminus K)$
  Zoberme si ľubovoľný z nespracovaných termov. Vieme, že každá premenná je v G. Takže pre nesplnenie podmienky by muselo $x_2 \in K$. Do K sa ale nedostala žiadna premenná, ktorá by nebola v nejakom spracovanom terme ako negovaná. Takže musela byť ešte v nejakom ďalšom terme ako negovaná. Vieme však, že $\forall e \neq e' : H_e(\mathcal{F}) \cap H_{e'}(\mathcal{F}) = \emptyset$. To je spor.
  Takže táto podmienka je splnená pre všetky ešte nespracované termy.

- $(x_1 \in K \vee x_1 \in O) \wedge (x \in K \vee x \in O)$
  Keď algoritmus označí nejaký term za spracovaný, premenná ktorá v ňom vystupuje ako negovaná skončí buď v K alebo v O alebo v P. Do bodu 6) algoritmu sa môžem dostať jedine z bodu 1) keď už nie je vo fronte P žiaden prvok, alebo z bodov 8), 7) ale v tomto prípade tiež nie je vo fronte P žiaden prvok (lebo do tohto bodu som sa dostal z bodu 6), kedy musela byť fronta P prázdna a ja som do nej nič nepridal). Takže vieme, že v momente rozhodovania sú všetky už spracované premenné (premenné ktoré

vystupujú ako negované v už spracovaných termoch) buď v K alebo v O.
Teraz tvrdím: medzi nespracovanými termami musí existovať aspoň jeden taký, ktorého obe pozitívne premenné sa už vyskytli v nejakom spracovanom terme ako negované. Ak by to tak totiž nebolo, znamenalo by to, že tieto nespracované termy sú "izolované". A to v tom zmysle, že sú odvoditeľné len z premenných, ktoré sú ako negované len v rámci tejto skupiny. Ak by v rámci ľubovoľnej DNF nejakej f_G (po príslušných modifikáciách) toto platilo, znamenalo by to, že by sa aspoň jeden vrchol e_S musel nachádzať v množine ich pozitívnych premenných. Toto ale v našom prípade nie je možné, pretože celú e_S sme už v indukčnom predpoklade zaradili do K.
Z oboch vyššie uvedených dostávame, že medzi nespracovanými termami musí byť aspoň jeden taký, že pre obe jeho pozitívne premenné x platí, že $x \in K \vee x \in O$.

$\square$

Kapitola 4

4-HM: NP-úplnosť

V tejto kapitole dokážem tvrdenie, ktoré by malo byť najväčším prínosom celej práce. A teda ukázať, že problém Hornovskej minimalizácie je NP-úplný vzhľadom k najpoužívanejšej miere $|\mathcal{F}|_t$, a to aj v prípade, že obmedzíme počet literálov v terme na štyri (v zhode so značením z prvej kapitoly budem používať skratku 4-HM-t).
Ako som už spomínal v úvode kapitoly 3, doteraz najsilnejším dokázaným výsledkom bolo, že problém Hornovskej minimalizácie je NP-úplný aj keď obmedzíme počet literálov na term na tri. Pre akýkoľvek vyšší počet literálov na term už tvrdenie platí automaticky (keďže je to len zovšeobecnenie problému). Avšak ukázal som, že v dôkaze tvrdenia je chyba, takže teraz nevieme nič nielen o zložitosti problému pri obmedzení na tri, ale ani na akýkoľvek vyšší počet literálov na term.
Takže dokázaním NP-úplnosti 4-HM bude táto novovzniknutá medzera aspoň čiastočne vyplnená. A kým sa neopraví predchádzajúci dôkaz, bude aj najsilnejším doteraz dokázaným tvrdením.
Kapitolu som rozdelil na dve časti. V prvej dokážem NP-úplnosť problému, ktorý som nazval 3-Set Covering. V druhej potom prevodom z tohto problému ukážem aj NP-úplnosť problému Hornovskej minimalizácie.

4.1 3-Set Covering

NP-úplnosť problému 3-Set Covering ukážem prevodom z problému nazvaného 3-Exact Cover. Avšak ani dôkaz tohto problému nebol nikde podrobne spracovaný, tak ho tu taktiež uvediem. A NP-úplnosť 3-Exact Cover ukážem prevodom z problému Tripartite Matching. Pod týmto problémom budem rozumieť:

Tripartite Matching:
Zadanie:

a) Tri množiny A, B, C, pričom $|A| = |B| = |C| = n$,

b) ternárnu reláciu $U \subseteq A \times B \times C$,

c) v U sú obsiahnuté všetky prvky každej množiny
(teda platí $\forall a \in A \exists u = (u_1, u_2, u_3) \in U$ tak, že $u_1 = a$, analogicky pre B a C).

Otázka: Existuje $W \subseteq U$ tak, aby $|W| = n$ a žiadne dve prvky W sa nezhodovali v žiadnej súradnici (tj. aby platilo $\forall (u_1, u_2, u_3), (v_1, v_2, v_3) \in W : u_1 \neq v_1 \wedge u_2 \neq v_2 \wedge u_3 \neq v_3$)?

Ak by sme v zadaní nemali podmienku c) a namiesto bodov a) a b) by sme mali položené $U \subseteq T \times T \times T, |T| = n$, potom by sme mali presne problém, ktorý ukázal ako NP-úplný už v roku 1972 Karp [10], kde ho nazval 3-DIMENSIONAL MATCHING. Ukážem, že zmeny, ktoré som v jeho definícii učinil sú zanedbateľné a NP-úplnosť tohto problému nijak nenarušujú.

Podmienka c).
Túto viac-menej iba technickú podmienku som si tam pridal, aby mi prevod na 3-Exact Cover sedel formálne presne. Je však vidieť, že je to podmienka celkom prirodzená. Ak by totiž nebola splnená, tak by sme mali okamžite odpoveď na Otázku kladenú problémom - bolo by to NIE. Dovolím si teda tvrdiť, že každá reálna implementácia algoritmu na jeho riešenie by si túto podmienku overila hneď na začiatku. To sa dá vykonať v čase $O(3|U|)$.
Ak by som mal načrtnúť presný formálny polynomiálny prevod z problému *bez* tejto podmienky na problém *s* touto podmienkou, potom by vyzeral nasledovne. Najprv v čase $O(3|U|)$ zisti, či sú množinou U pokryté všetky prvky každej množiny A, B, C.
Ak nie sú pokryté všetky prvky, potom preveď tento problém nasledovne: $A = B = C = \{0, 1\}$, $U = \{(0, 0, 0), (1, 1, 0), (0, 1, 1)\}$. Keďže neboli pokryté, problém bez podmienky c) by dal odpoveď NIE. Pri tomto zadaní da odpoveď NIE aj problém s touto podmienkou (jednoduché overiť rozborom - všetkých troch - možností).
Ak sú pokryté, preveď problém 1:1 - presne rovnaké zadanie.

Podmienka troch množín A, B, C namiesto jednej T.
O tom, že táto podmienka NP-úplnosti nijak nebráni, svedčí napríklad aj fakt, že NP-úplnosť Tripartite Matching v presne mojom zadaní, ale bez podmienky c), dokázal už aj Papadimitriou [13] (str. 199–201). A to prevodom priamo z 3SAT.
Ale to, že keď položím A, B, C namiesto jednej T tak to nijak nenarušuje pôvodné zadanie, je už na prvý pohľad jasné. Prvky U sú totiž usporiadané trojice a teda sa nijak "nemiešajú". Pri prevode z 3-Dimensional Matching na Tripartite Matching by teda stačilo položiť $A = B = C = T$. Problémy sú potom úplne ekvivalentné a nie je potreba robiť ani žiaden prevod.

4.1.1 3-Exact Cover je NP-úplný

V literatúre sa bežne objavuje problém s názvom Exact Cover, je to dokonca jeden z najstarších NP-úplných problémov, ukázal to už Karp [10]. Avšak na dokázanie NP-úplnosti problému 3-Set Covering budem potrebovať jeho modifikovanú verziu, ktorú som nazval 3-Exact Cover (3-EC). Je to problém identický s Exact Cover, avšak má obmedzenie mohutnosti množín v ňom participujúcich na 3. V tomto oddiele dokážem NP-úplnosť tohto problému.

3-Exact Cover:
Zadanie: Množina $X = \{x_1, \ldots, x_{3k}\}$ premenných, jej podmnožiny $S_1, \ldots, S_m$, kde $\forall j : S_j \subseteq X, |S_j| = 3$ a $\bigcup_{j=1\ldots m} S_j = X$.
Otázka: Existuje presne k množín $S_{j_1}, \ldots, S_{j_k}$ tak, aby $\bigcup_{i=1\ldots k} S_{j_k} = X$?

Najprv overím, či je tento problém v NP.

To je ale očividné. Majme certifikát, riešenie problému 3-EC, množiny $S_{j_1}, \ldots, S_{j_k}$. Najprv v čase $O(k)$ popočítam, či ich je skutočne presne k. Ak nie, certifikát zamietnem.
V čase $O(3k)$ zistím, či pokrývajú celú X.

NP-ťažkosť tohto problému ukážem prevodom z Tripartite Matching. Prevod je úplne priamočiary, keďže TM je špeciálnym prípadom 3-EC.

Tripartite Matching:
Zadanie: Tri množiny A, B, C, pričom $|A| = |B| = |C| = n$, ternárnu reláciu $U \subseteq A \times B \times C$, v U sú obsiahnuté všetky prvky každej množiny
Otázka: Existuje $W \subseteq U$ tak, aby $|W| = n$ a žiadne dve prvky W sa nezhodovali v žiadnej súradnici?

3-Exact Cover z neho skonštruovaná:
$X = A \cup B \cup C, k = n$, pre každé $u = (u_1, u_2, u_3) \in U$ skonštruujem $S_j = \{u_1, u_2, u_3\}$.

Teraz overím ekvivalenciu problémov. To je, že keď mám riešenie jedného, že prevod mi skutočne zaručil, že som tým dostal aj riešenie druhého.

TM $\Longrightarrow$ 3-EC
Na prvý pohľad je jasné, že podmienky, ktoré mi kladie zadanie na množiny S_j sú splnené a skutočne platí, že $\forall j : S_j \subseteq X, |S_j| = 3$ a $\bigcup_{j=1\ldots m} S_j = X$.
Majme W - riešenie problému TM. Riešenie problému 3-EC skonštruujem nasledovne: pre každý $w = (w_1, w_2, w_3) \in W$ polož $S_{j_i} = \{w_1, w_2, w_3\}$. Vieme, že žiadne dve prvky množiny W nemajú ani jednu súradnicu zhodnú. A keďže je týchto prvkov dohromady n, musia nutne pokrývať každú z množín A, B, C. Použili sme všetky prvky W, každý sme dali do nejakej S_{j_i}, takže zjednotenie množín S_{j_i} pokrýva zjednotenie množín $A \cup B \cup C$, čo je presne X. Zároveň $|W| = n \Rightarrow |\{S_{j_i}\}| = n = k$.

3-EC $\Longrightarrow$ TM
Majme riešenie problému 3-EC, množiny $S_{j_1}, \ldots, S_{j_k}$, ktoré pokrývajú celú X. Každá z množín $S_{j_i} = \{x_p, x_q, x_r\}$ obsahuje po jednom elemente z množín A, B, C, nech $x_p \in A, x_q \in B, x_r \in C$. Pre každú takúto množinu môžeme teda vytvoriť jeden prvok $w \in W$ nasledovne $w = (x_p, x_q, x_r)$. Počet prvkov W je teda $k = n$, pokrývajú všetky množiny A, B, C, z toho vyplýva že každý prvok W má jedinečnú každú súradnicu a teda žiadne dve nemajú súradnice rovnaké. $\square$

4.1.2 3-Set Covering je NP-úplný

Podobne ako v predchádzajúcom prípade, aj problém Set Covering je jeden z najstarších NP-úplných problémov [10]. Avšak na dokázanie kľúčového tvrdenia tejto kapitoly budem potrebovať opäť jeho modifikovanú verziu, ktorú som nazval 3-Set Covering (Množinové pokrytie s obmedzením mohutnosti množín na 3, skratka 3-SC). V tomto oddiele dokážem, že tento problém zostáva NP-úplný aj pri tomto obmedzení mohutnosti zdrojových množín. Z tohto problému potom odvodím NP-úplnosť problému Hornovskej minimalizácie. Najprv popíšem, ako znie zadanie problému.

3-Set Covering:
Zadanie: Množina $X = \{x_1, \dots, x_n\}$ premenných, jej podmnožiny $S_1, \dots, S_m$, kde $\forall j : S_j \subseteq X, |S_j| = 3$ a $\bigcup_{j=1\dots m} S_j = X$ a dané prirodzené číslo h.
Otázka: Existuje nanajvýš h množín $S_{j_1}, \dots, S_{j_h}$ tak, aby $\bigcup_{k=1\dots h} S_{j_k} = X$?

Náleženie do NP.
Očividné. Majme certifikát, riešenie problému 3-SC, množiny $S_{j_1}, \dots, S_{j_l}$. Najprv v $O(l)$ čase popočítam, či ich je skutočne maximálne h (teda či $l \leq h$). Ak nie, certifikát zamietnem. V čase $O(3l + n)$ zistím, či pokrývajú celú X.
$\square$

NP-ťažkosť tohto problému ukážem prevodom z problému z predchádzajúceho oddielu - 3-Exact Cover. Prevod je úplne priamočiary, keďže 3-EC je opäť špeciálnym prípadom 3-SC.

3-Exact Cover:
Zadanie: Množina $X' = \{x'_1, \dots, x'_{3k}\}$ premenných, jej podmnožiny $S'_1, \dots, S'_m$, kde $\forall j : S'_j \subseteq X', |S'_j| = 3$ a $\bigcup_{j=1\dots m} S'_j = X'$.
Otázka: Existuje presne k množín $S'_{j_1}, \dots, S'_{j_k}$ tak, aby $\bigcup_{i=1\dots k} S'_{j_k} = X'$?

3-Set Covering z neho skonštruovaný:
$X = X', n = 3k, (S_1 = S'_1, \dots, S_m = S'_m), h = k$

3-EC $\Longrightarrow$ 3-SC
Na prvý pohľad je jasné, že podmienky, ktoré mi kladie zadanie na množiny S_j sú splnené a skutočne platí, že $\forall j : S_j \subseteq X, |S_j| = 3$ a $\bigcup_{j=1\dots m} S_j = X$.
Majme riešenie problému 3-EC: Presne k množín $S'_{j_1}, \dots, S'_{j_k}$ tak, aby $\bigcup_{i=1\dots k} S'_{j_k} = X'$. Za riešenie problému 3-SC zvolím presne tie isté množiny, teda $S_{j_1}, \dots, S_{j_k}$. Viem, že mi pokrývajú celú X a je ich nanajvýš h (pretože ich je presne h).

3-SC $\Longrightarrow$ 3-EC
Majme riešenie problému 3-SC, množiny $S_{j_1}, \dots, S_{j_l}, l \leq h$, ktoré pokrývajú celú X. Keďže $n = 3k = 3h$, počet prvkov X je $3h$, potom nutne $l = h$. Takže máme presne k množín, ktoré pokrývajú celé X, riešením problému 3-EC sú teda skutočne množiny $S'_{j_1} = S_{j_1}, \dots, S'_{j_h} = S_{j_h}$.
$\square$

4.2 4-HM-t je NP-úplný

Použijem na to dôkaz, ktorým Ausiello [1] ukázal obdobné tvrdenie v oblasti hypergrafov prevodom zo Set Covering. Dôkaz však vyžadoval radu úprav. Najzávažnejšou bola modifikácia redukcie problému Set Covering na inštanciu - v jeho prípade "hypergrafu", - v mojom prípade Hornovskej DNF. Jeho dôkaz totiž dokazoval NP-úplnosť pre ľubovoľný hypergraf, neobmedzoval veľkosť hyperhrán - ekvivalent obmedzenia počtu literálov v terme. A tak v jeho dôkaze by ani použitie 3-Set Covering namiesto všeobecného Set Covering nepomohla.

Ďalším problémom bolo to, že v prípade hypergrafov si autor mohol dovoliť povedať len "táto hrana musí byť v každom hypergrafe". V prípade dokázania esenciálnosti termov je situácia o čosi zložitejšia.
Výsledkom tejto kapitoly bude dôkaz nasledujúcej vety.

Veta 4.1 *Problém Hornovskej minimalizácie v miere $|\mathcal{F}|_t$ je NP-úplný aj keď obmedzíme počet literálov v každom terme vstupnej DNF na 4.*

Predtým než prejdem ku dôkazu tejto vety, ukážem ako bude vyzerať prevod tohto problému a ukážem aj niekoľko dôležitých vlastností novovzniknutej inštancie Hornovskej DNF. S ich pomocou už bude dôkaz jednoduchý.

3-Set Covering:
Zadanie: Množina $X' = \{x'_1, \ldots, x'_n\}$ premenných, jej podmnožiny $S'_1, \ldots, S'_m$, kde $\forall j : S'_j \subseteq X', |S'_j| = 3$ a $\bigcup_{j=1 \ldots m} S'_j = X'$ a dané prirodzené číslo h.
Otázka: Existuje nanajvýš h množín $S'_{j_1}, \ldots, S'_{j_h}$ tak, aby $\bigcup_{k=1 \ldots h} S'_{j_k} = X'$?

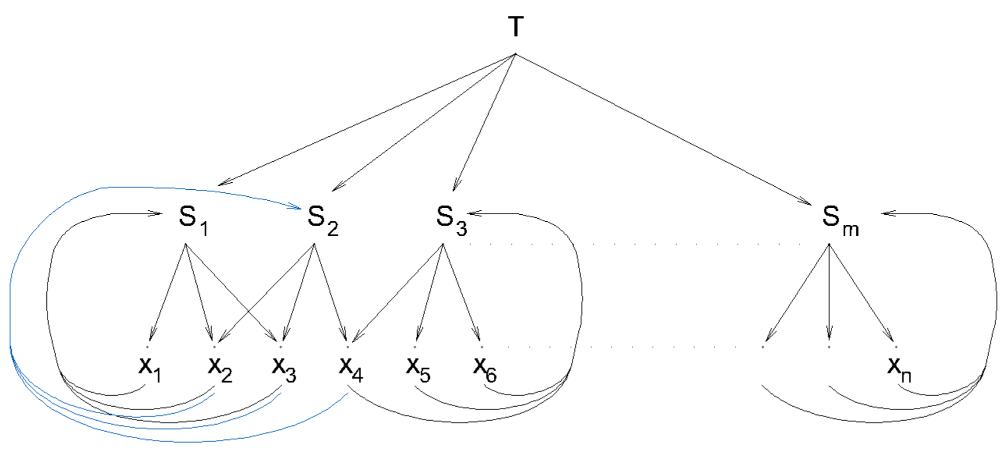
4-HM-t z neho skonštruovaná:
Množina premenných tohto HM bude pozostávať z $x_1, \ldots, x_n, S_1, \ldots, S_m$ a naviac premenná T. Pre každú zdrojovú množinu $S'_j, j = 1 \ldots m$, náležia do $\mathcal{F}$ tri typy termov

a) $T\bar{S}_j$

b) $S_j \bar{x}_i$ pre $\forall x'_i \in S'_j$

c) $x_{i_1} x_{i_2} x_{i_3} \bar{S}_j$ kde $x'_{i_1}, x'_{i_2}, x'_{i_3} \in S'_j$

Za premennú ohraničujúcu počet termov zvolíme $k = \sum_{j=1 \ldots m}(|S'_j|) + m + h$.



Obr. 4.1: Hypergraf vizualizujúci 4-HM-t asociovanú s inštanciou Set Covering

Hypergrafy (tj. grafy kde sú hrany nahradené hyperhranami, ktoré môžu vychádzať z viacero vrcholov a končiť v jednom) síce s problematikou priamo nesúvisia, ale na základe Obrázku 4.1 si môžme lepšie vytvoriť predstavu o tom, ako vyzerá skonštruovaná inštancia

4-HM-t. Jedna hyperhrana odpovedá jednému termu, vrcholy *z* ktorých ide hyperhrana odpovedajú pozitívnym literálom, vrchol *do* ktorého vedie hyperhrana zas negatívnemu literálu. “Chodenie” po hyperhranách grafu odpovedá postupu procedúry forward chaining. Ak začneme s nejakou množinou R premenných, tak z nej sa môžme vybrať po ľubovoľnej hyperhrane takej, že všetky vrcholy z ktorých hyperhrana vychádza sú v R. Vrchol, do ktorého táto hyperhrana vchádza môžme pridať do množiny R. Ak sa teda týmto spôsobom vieme dostať z množiny premenných R do premennej y, tak term $R\bar{y}$ je implikantom funkcie, ktorú tento graf reprezentuje.

Keďže v inštancii 3-Set Covering je obmedzená mohutnosť podmnožín S_j na 3, tak všetky termy majú nanajvýš 4 literály. Takže vzniknutá inštancia je skutočne 4-HM.
Ako neskôr popíšem v procese prevodu už zminimalizovanej inštancie 4-HM-t späť na inštanciu 3-Set Covering, toto získané riešenie pred samotným prevodom najprv (v polynomiálnom čase - Lemma 1.6) prevediem na primárnu irredundantú DNF. A až potom späť na inštanciu 3-Set Covering. Takže nižšie pri dokazovaní vlastností tejto získanej zminimalizovanej DNF môžem kľudne predpokladať, že je primárna irredundantná. Naproti tomu, $\mathcal{F}$ nie je irredundantná (vďaka čomu ju môžem vzhľadom k počtu termov minimalizovať). Avšak primárna je.

Primalita $\mathcal{F}$.
Overme si najprv, či sú všetky termy $\mathcal{F}$ primárne.
Pomerne jednoducho si môžeme overiť, že $\mathcal{F}$ nemá implikanty dĺžky jedna. Pre spor nech existuje taký pozitívny literál, ktorý je implikantom. Ohodnoťme jeho premennú hodnotou jedna. Potom by som si vynútil hodnotu nula pre funkciu (reprezentovanú DNF $\mathcal{F}$) tým, že by som ohodnotil aj všetky zvyšné premenné hodnotou jedna. Spor. To platí pre pozitívne literály zložené z ľubovoľnej premennej $\mathcal{F}$. Takže žiaden pozitívny literál nie je implikantom. Overím ešte implikanty dĺžky jedna vo forme negatívneho literálu. Nech teda $\bar{T} = 1$, potom položením $\forall j : S_j = 1$ a $\forall i : x_i = 1$ vynútim hodnotu funkcie 0. Ak $\bar{S}_j = 1$ alebo $\bar{x_i} = 1$, potom si hodnotu 0 pre funkciu vynútim, ak položím $T = 0$ a $\forall j : S_j = 0$ a aj $\forall i : x_i = 0$.
Nemám teda implikanty dĺžky jedna a z toho automaticky vyplýva, že termy z bodu a) a b) v definícii $\mathcal{F}$ sú primárne. Zostáva overiť termy definované v bode c). Nech pre spor existuje taký term $x_a x_b x_c \bar{S}_d$ v $\mathcal{F}$, ktorý nie je primárny. BÚNO nech x_a je premenná o ktorú sa dá skrátiť. Tvrdím teda, že $x_b x_c \bar{S}_d$ je implikant $\mathcal{F}$. Procedúra forward chaining so vstupom $\{x_b, x_c\}$ však k tejto množine už nič nepridá. A to preto, lebo jediné termy, ktoré obsahujú premenné x_i sú tie čo sú definované v bode c). Ale všetky tieto termy majú práve tri pozitívne premenné (keďže $|S_j| = 3, \forall j$), takže dve premenné vo vstupnej množine nemôžu “uspokojiť” žiaden term. A teda vstupná množina sa už neobohatí ani o jedinú premennú. A teda ani o premennú S_d.

Kvôli prehľadnosti si zavediem nasledujúce značenie. Pojmom *implikant typu X* budem označovať všetky primárne implikanty funkcie reprezentovanej DNF $\mathcal{F}$, ktoré neobsahujú premennú T a obsahujú niektorú z premenných x_i vo forme negatívneho literálu. Podobne *implikant typu S* bude označovať všetky primárne implikanty funkcie reprezentovanej DNF $\mathcal{F}$, ktoré neobsahujú premennú T a obsahujú niektorú z premenných S_j vo forme negatívneho literálu. Termy, ktoré obsahujú premennú T ma nateraz nebudú zaujímať.
A ešte jedno pomocné značenie: $I(j) = \{i | x'_i \in S'_j\}$.
Vieme, že konsenzuálna metóda generuje všetky primárne implikanty. A že každá primárna DNF sa skladá výlučne z primárnych implikantov. V nasledujúcom texte sa vo forme Lemmat

pokúsim ukázať, aké všetky možné primárne implikanty jednotlivých typov môžu existovať v nejakej primárnej DNF, ktorá bude ekvivalentná $\mathcal{F}$.

Lemma 4.2 *Neexistuje žiaden iný implikant typu X okrem tých, ktoré sú definované v bode b) definície $\mathcal{F}$.*

Dôkaz. Pre spor predpokladajme, že máme primárny implikant $P = (\bigwedge_{j\in J} S_j) \wedge (\bigwedge_{i\in I} x_i) \wedge \bar{x_k}$. Vždy keď $x'_i \in S'_j$, tak $S_j\bar{x_i} \in \hat{\mathcal{F}}$. Takže pre všetky $j \in J$ platí, že $x'_k \notin S'_j$, inak by to bol spor s primalitou P. Teraz vieme, že P je primárny implikant funkcie, ktorú reprezentuje aj $\mathcal{F}$. Takže musí byť odvoditeľný forward chainingom nad $\mathcal{F}$. Presne povedané forward chaining nad $\mathcal{F}$ so vstupnou množinou $\{S_j|j \in J\} \cup \{x_i|i \in I\}$ musí do výslednej množiny zahrnúť aj x_k. Lenže x_k sa môže pridať do výslednej množiny len pomocou nejakého S_j a to pomocou implikantu typu X. Vieme však, že medzi $\{S_j|j \in J\}$ taký term nie je. Nemôže sa však pomocou zvyšných x_i pridať? Nemôže, pretože by vo výslednej množine museli byť už predtým obsiahnuté všetky prvky jeho S'_j, takže aj x_k. To by ale znamenalo, že $k \in I$ (keďže x_k nie je v žiadnom $S_j, j \in J$), čo nie je možné - nemôže byť v terme aj premenná, aj jej negácia.
□

Lemma 4.3 *Každý implikant typu S je tvaru $P = (\bigwedge_{i\in I} x_i) \wedge (\bigwedge_{j\in J} S_j) \wedge \bar{S_l}$ a spĺňa nasledujúce podmienky:*

a) Pre každé $i' \in I(l) \setminus I$ existuje $j' \in J$ že $i' \in I(j')$.

b) $I(l) \nsubseteq I$

c) $I \subseteq I(l)$

d) $I \cap I(j) = \emptyset$ pre každé $j \in J$

e) $I = \emptyset \Rightarrow |J| \geq 2$

Dôkaz. Podmienka a).
Pre spor nech existuje také i' pre ktoré to neplatí. Začnem forward chaining nad $\mathcal{F}$so vstupnou množinou $\{x_i|i \in I\} \cup \{S_j|j \in J\}$. Za prvé: premenné S_j mi pridajú len $x_i \neq x_{i'}$. Za druhé: všetky x_i mi teraz pridajú len také S_j, ktoré neobsahujú $x_{i'}$. Tieto dve tvrdenia zostávajú v platnosti pre každý krok procedúry forward chaining. V konečnom kroku forward chaining skončí a $x_{i'}$ nebude pridané. Ak sa nepridá $x_{i'}$, nemôže sa pridať ani S_l. Spor s predpokladom, že P je implikant.
Podmienka b).
Inak by to bol spor s predpokladom, že P je primárny.
Podmienka c).
Ak by to nebola pravda, nech $I' = I \cap I(l)$. Na základe prvého bodu by bol potom implikantom aj term $(\bigwedge_{i\in I'} x_i) \wedge (\bigwedge_{j\in J} S_j) \wedge \bar{S_l}$. A to je spor s primalitou P.
Podmienka d).
Nech teda existuje také i', že $i' \in I$ a zároveň $x_{i'} \in S_{j'}$. Z bodu c) potom nutne aj $x_{i'} \in S_l$. Nuž ale v tomto prípade aj term $(\bigwedge_{i\in I\setminus\{i'\}} x_i) \wedge (\bigwedge_{j\in J} S_j) \wedge \bar{S_l}$ je implikantom $\mathcal{F}$. A to preto,

lebo forward chaining by v prvom kroku pridal na základe termu $S_{j'}\bar{x_{i'}}$ do výslednej množiny $x_{i'}$ a potom by pokračoval úplne rovnako ako pri terme P. Spor s primalitou P.
Podmienka e).
Ak by bolo veľkosti 1, potom by term $S_j\bar{S}_l$ bol implikantom. Pri forward chaining by sa z jedinej premennej S_j museli pridať tri premenné x_i tak, aby tie potom odvodili S_l. Ale to je možné jedine v prípade, že $S'_j = S'_l$. Predpokladajme však, že zadanie 3-Set Covering toto nepripúšťa. Predpokladať to môžme preto, lebo by sme mohli úplne rovnako robiť celý tento dôkaz pre modifikované 3-SC, v ktorom by bola podmienka, že žiadne dve množiny S_j nemajú presne tie isté literály. A toto modifikované 3-SC je tiež NP-úplné. Prevod z klasického 3-SC by bol priamočiary - presne rovnaké zadanie. Ak by sme našli riešenie v modifikovanom, bolo by to presne riešenie aj v pôvodnom. Ak by sme našli v pôvodnom, odstránili by sme duplikáty a bolo by to riešením aj v modifikovanom (možno lepším).
□

Dôsledok 4.4 *Nasledujúca tabuľka zhrňuje všetky možné primárne implikanty typu S, v ktorom je $\bar{S}_l$.*

$x_ix_jx_k\bar{S}_l$	$x_ix_jS_p\bar{S}_l$
$x_iS_p\bar{S}_l$	$x_iS_pS_q\bar{S}_l$
$S_pS_qS_r\bar{S}_l$	$S_pS_q\bar{S}_l$

Pričom $x_i, x_j, x_k \in S_l$ a zároveň $x_i, x_j \notin S_p, S_q$.

Teraz, keď viem ako vyzerajú všetky primárne implikanty funkcie reprezentovanej $\mathcal{F}$, som pripravený ukázať, že všetky implikanty, ktoré som uviedol v definícii $\mathcal{F}$ v bodoch b) a c) sú esenciálne. To znamená, že sa musia nachádzať vo všetkých primárnych DNF, ktoré reprezentujú tú istú funkciu ako $\mathcal{F}$. Term $P \in \mathcal{F}^*$ je esenciálny, práve keď nie je implikantom $\mathcal{F}^* \setminus P$, kde symbolom $\mathcal{F}^*$ označujem množinu všetkých primárnych implikantov funkcie reprezentovanej pomocou DNF $\mathcal{F}$.

Lemma 4.5 *Všetky implikanty typu X sú esenciálne.*

Dôkaz. Ukážeme pomocou forward chaining. Zoberme si $P = S_b\bar{x_a}$, chceme ho odvodiť v $\mathcal{F}^* \setminus P$. Takže začneme z množinou $\{S_b\}$. Keďže nemám žiadne x_i, tak na základe Lemma 4.3 e) viem, že implikant typu S použiť nemôžem. Použijem všetky implikanty typu X ktoré môžem, do výslednej množiny doplním $x_e, x_f \in S_b$. Takže v $R_{\mathcal{F}}(\{S_b\})$ mám $\{S_b, x_e, x_f\}$. Teraz však už nemôžem použiť ani žiaden implikant typu S, pretože mám len dve premenné x_i a my vieme, že $|S_j| = 3, \forall j$. Je tam síce aj S_b, ale vďaka Lemma 4.3 d) nenájdem žiaden primárny implikant, v ktorom by sa nachádzal spolu s x_e, x_f.
□

Lemma 4.6 *Všetky implikanty typu S uvedené v bode c) definície $\mathcal{F}$ sú esenciálne.*

Dôkaz. Podobne ako v predchádzajúcom dôkaze ukážeme pomocou forward chaining. Zoberme si $P = x_ix_jx_k\bar{S}_l$, chceme ho odvodiť v $\mathcal{F}^* \setminus P$. Forward chaining začne s množinou $\{x_i, x_j, x_k\}$ kde $x_i, x_j, x_k \in S_l$. Implikanty typu X sa použiť nedajú. Spomedzi všetkých implikantov typu S by sa dal použiť len jeden, a tým je P, ten však použiť nemôžme.

□

Z predchádzajúceho textu vyplynulo, že jediné termy, ktoré môžem pri minimalizácii z $\mathcal{F}$ odstrániť, sú tie ktoré obsahujú premennú T. A odstránim práve tie z nich, ktoré obsahujú také S_j, čo nevystupujú v riešení problému 3-Set Covering. Ako ukážem ďalej, vzniknutá DNF bude aj po ich odstránení ekvivalentná $\mathcal{F}$.

Teraz už mám všetky poznatky k tomu, aby som mohol ukázať, že inštancie jednotlivých problémov sú skutočne ekvivalentné.

Dôkaz Vety 4.7:
Náležitosť do NP som už ukázal pre obecné HM v Lemma 2.2.
4-HM-t je podproblém, takže náleženie do NP sa samozrejme zachováva.

3-SC $\Longrightarrow$ 4-HM-t
Majme teda riešenie problému 3-SC: $S'_{j_1}, \dots, S'_{j_h}$ tak, aby $\bigcup_{k=1\dots h} S'_{j_k} = X'$. Skonštruujem DNF $\mathcal{G}$, ktoré bude riešením 4-HM-t, takto: Do $\mathcal{G}$ dám všetky termy typu b) a c) v takom tvare ako sú v popise 4-HM-t. Spomedzi termov typu a) dám práve $T\bar{S_{j_1}}, \dots, T\bar{S_{j_h}}$. Na prvý pohľad je vidieť, že celkový počet termov v $\mathcal{G}$ je $\sum_{j=1\dots m}(|S'_j|) + m + h$. Je $\mathcal{G}$ ekvivalentné $\mathcal{F}$?

- $\mathcal{G} \leq \mathcal{F}$
  Každý term z $\mathcal{G}$ je prítomný aj v $\mathcal{F}$, takže $\mathcal{G} \leq \mathcal{F}$.

- $\mathcal{F} \leq \mathcal{G}$
  Jediné termy, ktoré sú v $\mathcal{F}$ a nie sú v $\mathcal{G}$ sú termy typu a). Pre nich overíme, či sú implikantami $\mathcal{G}$. Učiníme tak procedúrou forward chaining. Stačí mi ukázať, že ak začnem s T, tak vo výslednej množine budú všetky premenné S_j. V prvom kroku procedúra použije všetky $T\bar{S_{j_1}}, \dots, T\bar{S_{j_h}}$. Keďže vieme, že $\bigcup_{k=1\dots h} S'_{j_k} = X'$, tak v nasledujúcom kroku procedúra pomocou termov $S_{i_k}\bar{x}_i$ pridá všetky $x_1, \dots, x_n$. Teraz sa už môžu použiť aj všetky termy typu c) a do výstupnej množiny procedúry forward chaining zaradiť aj zvyšné S_j (ktoré sa nepridali už v prvom kroku).

4-HM-t $\Longrightarrow$ 3-SC
Majme teda DNF $\mathcal{G}'$, ktorá je ekvivalentná $\mathcal{F}$ a majúca maximálne $k = \sum_{j=1\dots m}(|S'_j|) + m + h$ termov. V polynomiálnom čase z nej vyrobíme ekvivalentnú primárnu irredundantú DNF $\mathcal{G}$ (Lemma 1.6). Vzhľadom ku spôsobu akým z nej robíme primárnu a irredundantnú DNF je jasné, že bude mať aj naďalej maximálne k termov (v tomto procese žiaden nepridávame). Na základe Lemma 4.5 a Lemma 4.6 vieme, že $\mathcal{G}$ má všetky termy, ktoré boli v definícii $\mathcal{F}$ v bodoch b) a c). A tých je $\sum_{j=1\dots m} |S'_j| + m$. Takže termov, ktoré môžu obsahovať T je nanajvýš h. Keďže musia byť primárne, môžu byť len v jednej z nasledujúcich dvoch kategórií:

i) $T\bar{S}_j$

ii) $T\bar{x}_i$

Pre každé $i \in \{1 \dots n\}$ platí, že $T\bar{x_i}$ je implikantom $\mathcal{F}$ (ľahké overiť - užitím forward chaining v dvoch krokoch). A keďže je $\mathcal{G}$ ekvivalentná $\mathcal{F}$, musia byť všetky tieto termy odvotiteľné pomocou forward chaining nad $\mathcal{G}$.

Teraz tvrdím: Pre každé $i \in \{1 \dots n\}$ platí, že buď sa x_i nachádza v terme kategórie ii), alebo je x_i' prvkom množiny S_j' takej, že S_j sa nachádza v terme kategórie i).
Ukážem to sporom. Predpokladajme, že existuje také x_a, ktoré nespĺňa ani jednu z týchto podmienok. Avšak aj pre toto x_a platí, že term $T\bar{x_a}$ je odvoditeľný v $\mathcal{G}$. Zoberme si priebeh procedúry forward chaining nad $\mathcal{G}$ so vstupnou množinou $\{T\}$. Nech teda P je term, s pomocou ktorého sme do výstupnej množiny vložili premennú x_a. Vieme, že v ňom musí x_a vystupovať ako negatívna, a že takéto termy sú v $\mathcal{G}$ len tvaru $S_j\bar{x_i}$ (keďže podľa predpokladu nespadá x_a do kategórie ii)). Nech teda $P = S_b\bar{x_a}$. Podľa predpokladu vieme, že S_b nie je v terme kategórie i). Takže sa musel v postupe procedúry forward chaining pridať nejakým termom $R = (\bigwedge_{j \in J} S_j) \wedge (\bigwedge_{i \in I} x_i) \wedge \bar{S_b}$. Keďže x_a sa pridalo v procedúre až neskôr, musí platiť $a \notin I$. Ale pre všetky termy typu S na základe Lemma 4.3 a) platí: pre každé $i' \in I(b) \setminus I$ existuje $j' \in J$ že $i' \in I(j')$. No ale keďže toto $S_{j'}$ je už vo výslednej množine procedúry forward chaining znamená to, že buď sa premenná x_a vložila skôr ako on (čo je spor s predpokladom), alebo tam bol zaradený už na začiatku tým, že bol v kategórii i). No ale to je taktiež spor s predpokladom tvrdenia.

Z práve dokázaného už priamo vyplýva, že ako riešenie problému 3-SC môžme položiť:

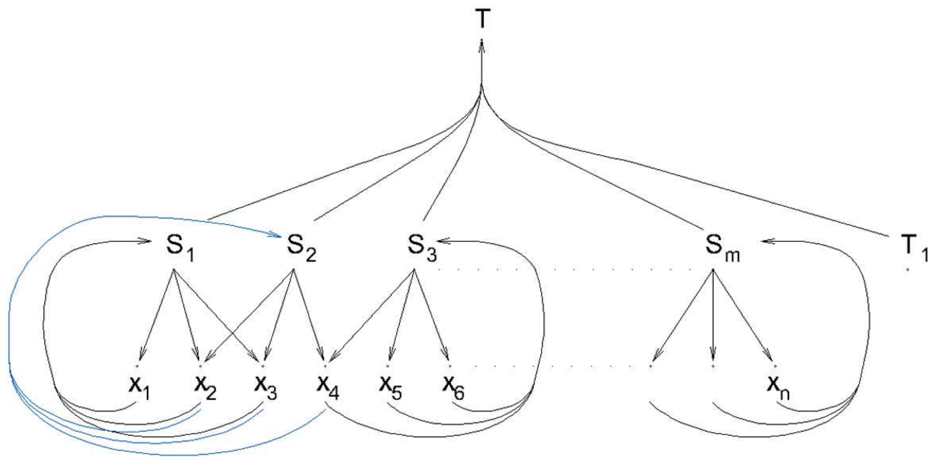
- všetky S_j, ktoré sa nachádzajú v terme kategórie i)
- pre každé x_i z termu kategórie ii) vyber náhodne jednu z S_j, ktorá ho obsahuje

Počet takýchto termov je v súčte menší než h, pokrývajú celú X', mohutnosti množín S_j som nemenil.
□

4.3 HM-a je NP-úplný

V tejto časti ukážem NP-úplnosť Hornovskej minimalizácie vzhľadom k miere $|\mathcal{F}|_a$. Tým doplním aj posledný chýbajúci článok kapitoly 2 a v obrázku 2.1 už budú dokázané nielen vzťahy medzi mierami, ale aj ich zložitosti.

Dôkaz spravím veľmi podobne, ako som v predchádzajúcej časti 4.2 ukázal NP-úplnosť Hornovskej minimalizácie vzhľadom k počtu termov. Prevod zo Set Covering sa len mierne pozmení. Ako ukazuje obrázok 4.2, pribudla jedna premenná T_1 a termy $T\bar{S}_j$ sa nahradia jediným termom $S_1 S_2 \dots S_m T_1 \bar{T}$. Vzhľadom k podobnosti dôkazov, uvediem tento len v skratke, vynechávajúc niektoré samozrejmé kroky.



Obr. 4.2: Hypergraf vizualizujúci HM-a asociovanú s inštanciou Set Covering

3-Set Covering:
Zadanie: Množina $X' = \{x'_1, \dots, x'_n\}$ premenných, jej podmnožiny $S'_1, \dots, S'_m$, kde $\forall j : S'_j \subseteq X', |S'_j| = 3$ a $\bigcup_{j=1\dots m} S'_j = X'$ a dané prirodzené číslo h.
Otázka: Existuje nanajvýš h množín $S'_{j_1}, \dots, S'_{j_h}$ tak, aby $\bigcup_{k=1\dots h} S'_{j_k} = X'$?

HM-a z neho skonštruovaná:
Množina premenných tohto HM bude pozostávať z $x_1, \dots, x_n, S_1, \dots, S_m$ a naviac premenné T a T_1. Pre každú zdrojovú množinu $S'_j, j = 1 \dots m$, náležia do $\mathcal{F}$ dva typy termov

b) $S_j \bar{x}_i$ pre $\forall x'_i \in S'_j$

c) $x_{i_1} x_{i_2} x_{i_3} \bar{S}_j$ kde $x'_{i_1}, x'_{i_2}, x'_{i_3} \in S'_j$

a k nim naviac jediný term

a) $P = S_1 S_2 \dots S_m T_1 \bar{T}$

Za premennú ohraničujúcu veľkosť zdrojovej plochy zvolíme $k = \sum_{j=1\dots m}(|S'_j|) + m + h + 1$. Označenie bodov definujúcich jednotlivé termy $\mathcal{F}$ som schválne prehádzal tak, aby sedeli s dôkazom 4-HM-t.

Veta 4.7 *Problém Hornovskej minimalizácie v miere $|\mathcal{F}|_a$ je NP-úplný.*

Dôkaz. Náležitosť do NP som už ukázal v Lemma 2.2.

3-SC $\Longrightarrow$ HM-a

Majme teda riešenie problému 3-SC: $S'_{j_1}, \ldots, S'_{j_h}$ tak, aby $\bigcup_{k=1\ldots h} S'_{j_k} = X'$. Skonštruujem DNF $\mathcal{G}$, ktoré bude riešením HM-a, takto: Do $\mathcal{G}$ dám všetky termy z bodov b) a c) tak, ako sú uvedené v definícii $\mathcal{F}$. Namiesto termu P dám term $P' = S_{j_1}S_{j_2}\ldots S_{j_k}T_1\bar{T}$. Na prvý pohľad je vidieť, že veľkosť zdrojovej plochy $\mathcal{G}$je $\sum_{j=1\ldots m}(|S'_j|)+m+h+1$. Je $\mathcal{G}$ekvivalentné $\mathcal{F}$?

- $\mathcal{F} \leq \mathcal{G}$
  Jediný term, v ktorom sa líšia je P. Chceme teda ukázať, že $P \leq \mathcal{G}$. Keď je $P = 1$, potom všetky jeho literály sú rovné jednej, o to viac ich podmnožina, takže z toho plynie, že aj term P' je rovný jednej. A teda aj $\mathcal{G}$.

- $\mathcal{G} \leq \mathcal{F}$
  Začneme procedúru forward chaining nad $\mathcal{F}$so vstupnou množinou $\{S_{j_1}, S_{j_2}, \ldots, S_{j_k}, T_1\}$. Keďže množiny S_j pokrývajú všetky premenné, v prvom kroku sa do výstupnej množiny pomocou termov b) zahrnú všetky premenné x_i. Z nich sa v druhom kroku odvodia všetky $S_j, j = 1\ldots m$. A z tohto už pomocou termu P sa zahrnie aj premenná T. Takže $P' \leq \mathcal{F}$. Ostatné termy sú zhodné.

HM-a $\Longrightarrow$ 3-SC

Majme teda DNF $\mathcal{G}'$, ktorá je ekvivalentná $\mathcal{F}$ a jej zdrojová plocha je maximálne $k = \sum_{j=1\ldots m}(|S'_j|) + m + h + 1$. V polynomiálnom čase z nej vyrobíme ekvivalentnú primárnu irredundantú DNF $\mathcal{G}$(Lemma 1.6). Jej zdrojová plocha sa vzhľadom na štruktúru primárnych implikantov nezväčší.

Keď sa pozrieme na všetky primárne implikanty, ktoré môžu z $\mathcal{F}$ vzniknúť konsenzuálnou metódou zistíme, že je situácia veľmi podobná tej z dôkazu 4-HM-t. Akurát v tomto prípade sa netvoria termy, ktoré obsahujú T ako pozitívny literál, iba negatívny. Takýchto termov je viac, vždy obsahujú aj premennú T_1 a niekoľko S_j a x_i.

Implikanty typu X a typu S majú teda presne rovnaký tvar. Implikanty obsahujúce T, T_1 mi nepomôžu ani tentoraz, preto aj v tomto prípade platia Lemma 4.5 a Lemma 4.6. Na základe nich vieme, že $\mathcal{G}$ má všetky termy, ktoré boli v definícii $\mathcal{F}$ v bodoch b) a c). Ich zdrojová plocha je v súčte $\sum_{j=1\ldots m}|S'_j| + m$ (keďže každé dve S'_j majú inú množinu prvkov). Takže zdrojová plocha zvyšných termov je nanajvýš $h + 1$. Medzi nimi sa nájde aspoň jeden s literálom $\bar{T}$ (T sa nemohlo "vytratiť").

Tvrdím: každý term z $\mathcal{G}$obsahujúci premennú T je tvaru $Q = (\bigwedge_{j\in J} S_j) \wedge (\bigwedge_{i\in I} x_i) \wedge T_1 \wedge \bar{T}$ a naviac platí, že $\forall i = 1\ldots n : (i \in I) \vee (\exists j' \in J : x'_i \in S'_{j'})$.

Sporom. Nech existuje také Q a také $a \in 1\ldots n$ pre ktoré to neplatí. Začneme forward chaining nad $\mathcal{F}$ s množinou $\{x_i|i \in I\} \cup \{S_j|j \in J\} \cup T_1$. Každé z $S_j, j \in J$ do výslednej množiny vloží všetky také x_i, že $x_i \in S_j$. Žiaden z nich teda nevloží x_a. Existuje aspoň jedno $S_r, r = 1\ldots m$ také, že $x_a \in S_r$. Toto S_r nebolo v počiatočnej množine ($r \notin J$) a zároveň sa nikdy do výslednej množiny ani pridať nemôže, keďže v nej nie je ani x_a. A teda sa nemôže v $\mathcal{F}$ použiť term P na odvodenie T. Takže Q nie je implikantom funkcie reprezentovanej pomocou $\mathcal{F}$ a teda $\mathcal{F}$ a $\mathcal{G}$ nie sú ekvivalentné. Spor.

Zoberme si ľubovoľné také Q. Z práve ukázaného tvrdenia vyplýva, že ako riešenie problému 3-SC môžme položiť:

- všetky S_j, ktoré sa nachádzajú v terme Q (teda $j \in J$)
- pre každé x_i z termu Q (teda $i \in I$) vyber náhodne jednu z $S_j, j = 1 \dots m$, ktorá ho obsahuje

Počet literálov v terme Q je nanajvýš $h + 1$, z toho minimálne jeden je T_1, takže týchto množín je nanajvýš h, pokrývajú celú X', mohutnosti množín S_j som nemenil.
$\square$

Kapitola 5

Záver

Hlavným účelom práce bolo podať prehľad o zložitosti problému hľadania minimálnej reprezentácie Hornovských funkcií so zvláštnym zreteľom na reprezentáciu pomocou DNF - problém nazývaný Horn Minimization. V odbornej literatúre sa pod týmto názvom väčšinou objavuje minimalizácia dĺžky DNF vzhľadom k počtu termov alebo literálov.
V druhej kapitole som sa pokúsil na základe výsledkov z teórie hypergrafov rozšíriť spôsob nazerania na dĺžku formule o čo najväčší počet mier, ktoré majú v prípade Hornovských funkcií dobrý zmysel. Ukázal som, že mnohé z nich spolu súvisia a to do takej miery, že sa každá Hornovská funkcia dá zminimalizovať vzhľadom ku všetkým mnou definovaným mieram (až na $|\mathcal{F}|_l$). Zároveň som ukázal zložitosť problému Hornovskej minimalizácie pre každú z nich. Otvoreným problémom zostala pozícia HM-l (počet literálov) vzhľadom k ostatným mieram. V práci sa neobjasnilo, či je implikovaná alebo implikuje inú minimálnu formu.
V tretej kapitole som podal podrobný rozbor dôkazu z článku [4], podľa ktorého zostáva Hornovská minimalizácia vzhľadom k počtu termov NP-úplná aj pri obmedzení dĺžky termov na tri (3-HM-t). Ukázal som, kde je v dôkaze chyba. Zároveň som sa pokúsil množstvo užitočných tvrdení v tomto dôkaze uvedených použiť na opravenie tohto dôkazu. Zadefinoval som problém HV-pokrytia, ktorý je ekvivalentný tomu z článku. Avšak zložitosť tohto problému som nepreukázal. Otvoreným problémom teda zostáva, či sa tento dôkaz dá opraviť. Napríklad tým, že by sa dokázala NP-úplnosť problému HV-pokrytia. Prípadne ukázať NP-úplnosť problému 3-HM-t iným spôsobom.
Záverečná kapitola potom priniesla najväčší prínos, keď medzeru vzniknutú nájdenou chybou v tretej kapitole zaplnila o čosi slabším tvrdením - totiž že Hornovská Minimalizácia vzhľadom k počtu termov zostáva NP-úplná aj pri obmedzení dĺžok termov na štyri. Dôkaz som učinil prevodom z problémov (3-Set Covering, 3-Exact Cover), ktorých NP-úplnosť som taktiež ukázal.
Zároveň som ukázal NP-úplnosť Hornovskej minimalizácie aj pre mieru doteraz u Hornovských funkcií neuvažovanú - o veľkosť zdrojovej plochy, čo je súčet *mohutností* zdrojových množín. Zaujímavým faktom je, že Hornovská minimalizácia vzhľadom k *počtu* zdrojových množín je ľahká.

Literatúra

[1] G. Ausiello, A. D'Atri, D. Saccà: *Minimal representation of directed hypergraphs*, SIAM J. COMPUT., Vol. 15, No. 2 (Máj 1986), 418-431

[2] A. Bhattacharya, B. DasGupta, Gy. Tur´an: *On approximate Horn minimization*. Pripravuje sa.

[3] Endre Boros, Ondřej Čepek, Alexander Kogan, Petr Kučera: *A Subclass of Horn CNFs Optimally Compressible in Polynomial Time*, RRR 11-2009 (Jún 2009)

[4] Endre Boros, Ondřej Čepek: *On the Complexity of Horn Minimization*, DIMACS Technical Report 94-7 (Máj 1994)

[5] Ondřej Čepek: *Booleovské funkce a jejich aplikace*, prednáška + osobná komunikácia

[6] Ondřej Čepek, Petr Kučera: *On the Complexity of Minimizing the Number of Literals in Horn Formulae*, RRR 11-2008

[7] Peter L. Hammer, Alexander Kogan: *Horn functions and their DNFs*, Information Processing Letters, 44(1992), 23-29

[8] Peter L. Hammer, Alexander Kogan: *Optimal Compression of Propositional Horn Knowledge Bases: Complexity and Approximation*, Artificial Intelligence, 64(1993), 131-145

[9] Peter L. Hammer, Alexander Kogan: *Quasi-acyclic propositional Horn knowledge bases: Optimal compression.*, IEEE Transactions on Knowledge and Data Engineering 7 (1995), 751-762.

[10] Richard M. Karp: *Reducibility among combinatorial problems*, Complexity of Computer Computations (edited by J. W. Thatcher and R. E. Miller), Plenum Press, New York (1972)

[11] David Kronus: *Minimální formy Hornovských funkcí*, Diplomová práce (2003)

[12] David Maier: *Minimum Covers in the Relational Database Model*, Journal of the Association for Computing Machinery, 27(1980), 664-674

[13] Christos H. Papadimitriou: *Computational Complexity*, Addison Wesley (1994)