Katedra Informatiky
Fakulta Matematiky, Fyziky a Informatiky
Univerzita Komenského, Bratislava

Komunikačná zložitosť

(Diplomová práca)

Autor: Bc. Miroslav Baláž

Školiteľ: Doc. RNDr. Pavol Ďuriš, CSc.

Bratislava, 2009

Čestne prehlasujem, že som diplomovú prácu vypracoval samostatne s použitím uvedenej literatúry. _____

Abstrakt: V práci sa zaoberáme komunikačnou zložitosťou *2-CNF* funkcií. Dokážeme existenciu lineárne ťažkej *2-CNF* funkcie. A popíšeme konkrétnu *2-CNF* funkciu, ktorej komunikačná zložitosť je $\Omega(\sqrt{n})$.

Kľúčové slova: Komunikačná zložitosť, *2-CNF*.

Abstract: In this text we prove lower bounds of communication compexity of *2-CNF* functions. We prove existence of *2-CNF* function with linear communication complexity and we describe concrete *2-CNF* function with communication complexity $\Omega(\sqrt{n})$.

Keywords: Communication complexity, *2-CNF*.

Obsah

Zoznam obrázkov

Kapitola 1

Úvod

Komunikačná zložitosť je asi 25 rokov stará oblasť informatiky. Zaoberá sa meraním množstva informácie, ktorú si musia vymeniť viacerí účastníci, aby sa na niečom dohodli alebo aby vypočítali nejakú funkciu. Autorom komunikačnej zložitosti je Yao, ktorý vymyslel prvé modely, poukázal na prvé problémy a priniesol prvé výsledky. V súčastnosti existuje viacero modelov komunikačnej zložitosti, najväčšie uplatnenie má komunikačná zložitosť pri hľadaní dolných odhadov pre paralelné výpočtové zariadenia. V tejto práci sa venujeme základnému modelu *dvojúčastníckej komunikačnej zložitosti* a prichádzame s dvomi nezávislými vlasntými výsledkami. Prvý hovorí o modifikácii základného modelu, v ktorej jeden z účastníkov má o trochu viacej informácie ako druhý. Druhý výsledok sme dosialhli pri skúmaní komunikačnej zložitosti *2-CNF funkcií*. Dokázali sme, že existuje *2-CNF funkcia* s lineárnou komunikačnou zložitosťou a našli sme konkrétnu *2-CNF funkciu* s komunikačnou zložitosťou $\Omega(\sqrt{n})$.

Rozčlenenie práce je nasledovné: V druhej kapitole zadefinujeme základné pojmy a model komunikačnej zložitosti, ktorým sa zaoberáme. V tretej kapitole uvádzame základné výsledky a aplikácie komunikačnej zložitosti, na niektoré výsledky z tejto kapitoly sa odvolávame aj vo vlastných výsledkoch. V štvrtej kapitole prezentujeme výsledky, ktoré sme dosiahli pri skúmaní $g(n)$-nevyváženého rozdelenia vstupu. V piatej kapitole sa venujeme komunikačnej zložitosi *2-CNF funkcií* a prezentujeme vlastné tvrdenia aj s dôkazmi.

Kapitola 2

Komunikačná zložitosť

V tejto kapitole zavedieme väščinu pojmov a označení, ktoré budeme v práci používať. Pre účel jednoznačnosti definujeme aj základné pojmy z teórie formálnych jazykov a z teórie grafov, z ktorých najdôležitejšie pojmy sú *bit*, *slovo*, *jazyk*. Kľúčovým pojmom popísaným v tejto kapitole je *Dvojúčastnícka zložitosť* a *Vyvážené premenlivé rozdelenie vstupu*.

2.1 Abeceda, slová, jazyky . . .

Definícia 2.1.1. Abeceda je konečná neprázdna množina slov.

V tejto práci budeme používať iba dvojprvkovú abecedu, tou bude množina $\{0, 1\}$, jej prvky budeme nazývať *bity*.

Definícia 2.1.2. Slovom nad abecedou Σ nazývame konečnú postupnosť, ktorej každý prvok je z množiny Σ.

Definícia 2.1.3. Dĺžka slova w je dĺžka postupnosti, ktorou je tvorené. Dĺžku slova označujeme ako $|w|$.

Prázdne slovo označujeme ako ϵ a jeho dĺžka je 0.

Definícia 2.1.4. Ak w je slovo, potom $w(i)$ alebo w_i označuje i-ty bit slova w, je to zároveň i-ty člen postupnosti, ktorá tvorí slovo w, ak prvky číslujeme od 0.

Definícia 2.1.5. Ak u a v sú slová, potom zreťazením týchto dvoch slov nazývame slovo w, pre ktoré platí: $w_i = u_i$ ak $i < |u|$, $w_i = v_{i-|u|}$ ináč. Zreťazenie slov u a v skrátene zapisujeme ako uv.

Definícia 2.1.6. Reverzom slova $w = w_0 w_1 \dots w_{n-1}$ je slovo $w^R = w_{n-1} w_{n-2} \dots w_1 w_0$.

Definícia 2.1.7. Jazyk je množina slov.

Jazyky väčšinou označujeme ako L s nejakým dolným indexom.

Definícia 2.1.8. Ak L a L' sú jazyky, tak zreťazením týchto jazykov je jazyk $L'' = \{uv | u \in L, v \in L'\}$

Definícia 2.1.9. $L^0 = \{\epsilon\}$

Definícia 2.1.10. $L^n = LL^{n-1}$ pre všetky n väčšie ako 1

Definícia 2.1.11. $L^* = \bigcup_{i>=0} L^i$

Definícia 2.1.12. Nech $f\colon \{0,1\}^* \to \{0,1\}$, potom $L_f = \{w | f(w) = 1\}$

Definícia 2.1.13. Komplement jazyka L je $L^C = \{0,1\}^* - L$

2.2 Čísla

Každému slovu nad abecedou $\{0,1\}$ možno priradiť prirodzené číslo. Pre toto priradenie zavedieme označenie, lebo ho budeme z technických príčin používať.

Definícia 2.2.1. Nech $w = w_0 w_1 \dots w_n$, potom $\mathrm{BIN}(w) = \sum_{i=0}^{n} w_i 2^{n-i}$.

Definícia 2.2.2. Nech x je prirodzené číslo. $\mathrm{BIN}_k^{-1}(x)$ je také slovo w, ktorého dĺžka je k a $\mathrm{BIN}(w)$ je x.

2.3 Grafy

V práci používame jazyk teórie grafov, predpokladáme, že čitateľ použité pojmy a výsledky bude poznať. Spomenieme spôsob, ktorý sa používa na reprezentovanie grafov, keď sa skúma komunikačná zložitosť nejakého grafového problému. Graf majúci n vrcholov budeme reprezentovať slovom dĺžky $\binom{n}{2}$, kde hodnota bitu na i-tej pozícii bude určovať, či je v grafe prítomná hrana číslo i pre nejaké poradie hrán.

2.4 Komunikačný model

Podľa [EK97], všeobecný komunikačný problém možno popísať nasledujúcimi slovami: Máme systém, ktorý musí vykonávať nejakú úlohu, ktorá závisí od informácií, ktoré sú rozdistribuované medzi rozličné časti systému

zvané účastníci. Účastníci teda musia medzi sebou navzájom komunikovať za účelom vykonania nejakej úlohy.

Model, ktorým sa budeme zaoberať v tejto práci (jeho autorom je Yao), sa snaží popísať najjednoduchšie možné scenáre komunikačného problému. Yaov model sa od všeobecného problému líši nasledovnými zjednodušujúcimi predpokladmi:

- V systéme sú iba dvaja účastníci.
- Každý z účastníkov dostane dopredu danú fixnú časť vstupu.
- Jediná meraná veličina je počet prenesených *bitov* v komunikácii.
- Úloha účastníkov je výpočet dopredu zadanej funkcie, ktorej parametrom je celý vstup.

Poznámka. Z tretieho prepokladu je teda dané, že účastníci majú neobmedzenú výpočtovú kapacitu.

Tieto predpoklady nám pomáhajú sústrediť sa na jadro problémov spojených s komunikáciou. Napriek zjavnej jednoduchosti je tento model veľmi bohatý, odhaľuje peknú štruktúru a dajú sa v ňom študovať veci akými sú randomizácia, nedeterminizmus a mnohé iné. Poznatky, ktoré nadobudneme v tomto modely, sa dajú preniesť do viacerých iných scenárov, v ktorých je kľúčová komunikácia.

2.4.1 Dvojúčastnícka zložitosť

Hore popísaný model sa pokúsime popísať formálne, ale najskôr ešte raz popíšeme o čo má ísť. Nech X, Y, Z sú ľubovoľné konečné množiny a nech $f\colon X \times Y \to Z$ je ľubovoľná funkcia. Máme dvoch účastníkov A a B, ktorí chcú vypočítať funkciu $f(x, y)$ pre nejaké vstupy $x \in X$ a $y \in Y$. Problém je v tom, že A pozná iba x a B pozná iba y. To znamená, že pre netriviálne funkcie musia spolu nejako komunikovať. Táto komunikácia bude uskutočnená podľa pevného protokolu $\mathcal{P}$ (ktorý zavisí iba od funkcie f a nie od vstupov) a bude prebiehať tak, že vždy jeden účastník pošle nejaké *bity* druhému účastníkovi, až kým jeden z účastníkov bude môcť jednoznačne určiť $f(x, y)$. Protokol $\mathcal{P}$ nám musí teda v každom kroku vedieť určiť nasledovné veci:

- Či už je komunikácia ukončená.
- Ak je komunikácia ukončená, aký je výsledok.
- Ak nie je komunikácia ukončená, tak ktorý účastník má poslať nasledovný *bit*.

- Aký *bit* má poslať účastník ktorý je na rade.

To, kto v danom momente posiela *bit*, musí byť určené iba dovtedajšou komunikáciou (prenesenými *bitmi*). A aký *bit* daný účastník pošle musí byť závislé iba na dovtedajšej komunikácii a na časti vstupu, ktorý pozná.

Keďže nepotrebujeme protokol vykonávať, ale stačí nám ho iba vedieť popísať, môžeme hore uvedené požiadavky zhrnúť do nasledovnej definície.

Definícia 2.4.1. [EK97] Protokol $\mathcal{P}$ definovaný nad $X \times Y$ s oborom hodnôt Z, je binárny strom, v ktorom každému vnútornemu vrcholu v je priradená buď funkcia $a_v \colon X \to \{0,1\}$ alebo $b_v \colon Y \to \{0,1\}$, pričom rôznym vnútorným vrcholom môže byť priradená rôzna funkcia a každému listu je priradená jedna hodnota zo Z. Hodnota protokolu P na vstupe (x,y) je hodnota listu, ktorý dosiahneme, keď začneme z koreňa prechádzať stromom a v každom vnútornom vrchole v označenom a_v pôjdeme doľava, keď $a_v(x) = 0$ a doprava, keď $a_v(x) = 1$ a vo vrchole označenom b_v pôjdeme odľava, vtedy keď $b_v(y) = 0$ a doprava, vtedy keď $b_v(y) = 1$. Cena protokolu $\mathcal{P}$ na vstupe (x,y) je dĺžka cesty, ktorú prejdeme pri výpočte na vstupe (x,y). Cena protokolu je hĺbka stromu, ktorým je tvorený.

Poznámka. V tejto práci sa ďalej predpokladá, že $X = \{0,1\}^n$, $Y = \{0,1\}^n$ a $Z = \{0,1\}$, pre nejaké n.

Intuitívne povedané, každý vnútorný vrchol v označený a_v zodpovedá situácii, keď má bit poslať účastník A a hodnota tohto bitu je $a_v(x)$. Každý vnútorný vrchol v označený b_v znamená že *bit* posiela účastník B a jeho hodnota je $b_v(y)$.

Definícia 2.4.2. [EK97] Pre funkciu $f \colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, deterministická komunikačná zložitosť f je cena najlacnejšieho protokolu, takého, že $f(x,y) = \mathcal{P}(x,y)$, pre všetky $x \in X$ a $y \in Y$. Túto hodnotu značíme ako $D(f)$.

2.4.2 Vyvážené premenlivé rozdelenie vstupu

Dôvod vysokej komunikačnej zložitosti funkcie f je niekedy iba v tom, ako rozdelíme vstup medzi dvoch účastníkov.

Pri skúmaní integrovaných obvodov sa prišlo na to, že existuje vzťah medzi veľkosťou obvodu a komunikačnou zložitosťou funkcie ktorú daný obvod počíta, ale pri takom rozdelení vstupu, ktorý túto zložitosť minimalizuje [Hro97]. Aby sme mohli ku komunikačnej zložitosti pristupovať aj takýmto spôsobom, zadefinujeme si nasledujúcu notáciu

Definícia 2.4.3. $v(f,i)$. Ku každej funkcii $f\colon \{0,1\}^{2n} \to \{0,1\}$ a
$I \subset \{0,1,\ldots,2n-1\} = (l_0, l_1, \ldots, l_{m-1})$,
$K = \{0,1,\ldots,2n-1\} \setminus I = (k_0, k_1, \ldots, k_{2n-m-1})$ možno priradiť funkciu $v(f,I)\colon \{0,1\}^{|I|} \times \{0,1\}^{2n-|I|}$ takú, že $v(f,I)(x,y) = f(z)$, kde

$$z(l_i) = x(i) \text{ pre všetky i od 0 do } m-1 \text{ a}$$
$$z(k_i) = y(i) \text{pre všetky i od 0 do } 2n-m-1$$

Zjednodušene povedané, $v(f,I)$ predstavuje takú funkciu, v ktorej prvý počítač dostane bity s pozíciami v I, a druhý počítač ostatné bity. Pomocou tejto notácie zadefinujeme ďalšie miery zložitosti.

Definícia 2.4.4. $D^{\mathrm{BEST}}(f) = \min_{|I|=n} D(v(f,I))$

Definícia 2.4.5. $D^{\mathrm{WORST}}(f) = \max_{|I|=n} D(v(f,I))$

Kapitola 3

Základné výsledky

3.1 Základné poznatky

V tejto kapitole uvádzame základné známe výsledky z teórie komunikačnej zložitosti a známe vety, ktoré používame vo vlastných tvrdeniach. Z nich sú najdôležitejšie vety o existencii ťažkých funkcií a metódy pre dolné odhady.

Najjednoduchší spôsob, ako môžu účastníci vypočítať každú funkciu, je taký, že účastník A pošle účastníkovi B celý svoj vstup. Účastník B už teda pozná aj x aj y a môže teda sám vypočítať $f(x, y)$.

Veta 3.1.1. *[EK97]* $D(f) \leq n$

Najjednoduchšia funkcia, pre ktorú lepší spôsob ani neexistuje je tá, ktorá potrebuje overiť, či sú vstupy, ktoré dostali účastníci, rovnaké.

Definícia 3.1.1. $f_{\mathrm{EQU}} \colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}, (f_{\mathrm{EQU}}(x,y) = 1) \iff x = y$

Veta 3.1.2. *[EK97]* $D(f_{\mathrm{EQU}}) = n$

Táto funkcia má aj tú vlastnosť, že hoci jej komunikačná zložitosť je vysoká, existuje také rozdelenie vstupu, pri ktorom je komunikačná zložitosť konštantná.

Veta 3.1.3. *[Hro97]* $D^{\mathrm{BEST}}(f_{\mathrm{EQU}}) = 2$

Navyše, podľa môjho názoru ide o funkciu, ktorá sa v praxi používa veľmi často. Predstavme si prípad, keď chceme overiť, či sú dva súbory na vzdialených počítačoch rovnaké. Z vety 3.1.2 vyplýva, že neexistuje deterministické riešenie, ktoré by bolo lepšie, ako poslať celý súbor druhému počítaču a ten ich porovná. V praxi sa však táto úloha dá riešiť jednoducho pomocou šifrovacích metód. Prvý počítač vypočíta akúsi šifrovaciu funkciu celého svojho

súboru a tým dostane reťazec, ktorý je krátky, ale verí sa, že je malá pravdepodobnosť, že ľudia vedia zostrojiť iný súbor s rovnakým výsledkom tejto funkcie. Takéto riešenie ale nefunguje vo všetkých prípadoch, na to, aby sme takéto riešenie mohli nájsť aj v teoretickej rovine, stačí nám zaviesť randomizáciu.

Randomizáciu môžeme v súvislosti s komunikačnou zložitosťou zaviesť rôznymi spôsobmi a môžeme skúmať viacero mier, tak ako je to aj v klasickej teórii výpočtovej zložitosti. V práci sa randomizovanými protokolmi nezaobárame, ale len pre ilustráciu ukážeme jeden zo spôsobov.

Tak ako sme v definícii *dvojúčastníckej zložitosti* používali strom, ktorého vnútorné vrcholy boli označené funkciou závislou na vstupe účastníka, v randomizovanom protokole budú tieto funkcie závislé od vstupu počítača a od náhodného reťazca, ktorý môže byť rôzny pre jednotlivých účastníkov.

Nech $R(f)$ je cena takéhoto protokolu s tým, že ak výsledok f je 0, tak pravdepodobnosť zlého výsledku je najviac $1/2$ a ak výsledok f je 1, tak protokol vždy dá výsledok 1.

Veta 3.1.4. *[EK97]* $R(f_{\mathrm{EQU}}) = O(\log n)$

Dôkaz. [EK97] Nech $p \geq n^2$ je prvočíslo. Na vstupy, ktoré dostanú účastníci, sa môžeme dívať ako na polynómy nad poľom $GF(p)$.

- $A(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}$
- $B(x) = b_0 + b_1 x + b_2 x^2 + \cdots + b_{n-1} x^{n-1}$

kde $a = a_0 a_1 \ldots a_{n-1}$ a $b = b_0 b_1 \ldots b_{n-1}$ sú vstupy, ktoré dostanú účastníci A, resp. B.

Účastník A zvolí náhodné číslo t menšie ako p a účastníkovi B pošle hodnoty t a $A(t)$. Ten vypočíta $B(t)$ a ak sa $B(t) = A(t)$, vyhlási výsledok, že $a = b$. Zjavne tento protokol môže dať zlý výsledok iba v prípade že $a \neq b$, ale pravdepodobnosť takejto udalosti je najviac $1/n$, lebo hodnôt v ktorých sa dva rôzne polynómy stupňa $n-1$ môžu rovnať je najviac $n-1$, sú to korene polynómu $A(x) - B(x)$. □

Prípadov, kedy neexistuje lepší spôsob ako poslať druhému účastníkovi celý svoj vstup je veľmi veľa. Nasledujúca veta hovorí o tom, že väčšina funkcií je takýchto.

Poznámka. Ďalej pojednávame o deterministickej komunikačnej zložitosti.

Veta 3.1.5. *[Hro97] Pravdepodobnosť, že náhodne zvolená funkcia* $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *bude mať komunikačnú zložitosť menej ako* n $(D(f) < n)$ *sa blíži k* 0 *s rastúcim* n.

Poznámka. Funkciu môžeme *náhodne* zvoliť tak, že náhodne priradíme funkčnú hodnotu pre každý vstup.

Keďže tých funkcií je veľa uvedieme zoznam niektorých z nich, nech $x = x_0x_1 \dots x_{n-1}$ a $y = y_0y_1 \dots y_{n-1}$:

- $x = y$ [EK97]
- $x > y$ [EK97]
- $\sum_{i=0}^{n-1} x_i y_i \mod 2$ [EK97]
- $(\sum_{i=0}^{n-1} x_i y_i) > 0$ [EK97]

Podobná situácia je aj keď zoberieme do úvahy *Vyvážené premenlivé rozdelenie.*

Veta 3.1.6. *[Hro97] Pre každé n existuje funkcia $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, taká že $D^{BEST}(f) = n$.*

Napriek faktu uvedenému v predchádzajúcej vete, nebola doteraz nájdená žiadna konkrétna funckia, pre ktorú by platilo $D^{\mathrm{BEST}}(f) = n$. Poznáme však funkcie pre ktoré poznáme lineárny dolný odhad pre D^{BEST}. Jednou z nich je aj funkcia uvedená v nasledujúcej definícii.

Definícia 3.1.2. Nech $2n = \binom{m}{2}$, $f_{\mathrm{tri}}\colon \{0,1\}^{2n} \to \{0,1\}$. $f_{\mathrm{tri}}(x) = 1$ práve vtedy, keď graf daný reprezentáciou x obsahuje trojuholnik.

Dôkaz, že $D^{\mathrm{BEST}}(f_{\mathrm{tri}}) \in \Omega(n)$ je uvedený v [Hro97].

3.2 Dolné odhady dvojúčastníckej zložitosti

Metód na určenie dolných odhadov komunikačnej zložitosti bolo zatiaľ objavených vicero a vzťahy medzi nimi nie sú vždy úplne známe. V tejto práci budeme používať iba jednu z nich, zvanú *klamúca množina* (fooling-set), aby sme sa k tomuto pojmu dopracovali, musíme zaviesť ďalšie pojmy, označenia a vzťahy medzi nimi.

Keď máme skúmať nejakú funkciu $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, je možné zapísať si funkčné hodnoty do matice rozmeru $2^n \times 2^n$.

Definícia 3.2.1. Nech pre funkciu $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, značíme označením M_f, maticu rozmeru $2^n \times 2^n$ takú, že $M_f(i,j) = f(\mathrm{BIN}_n^{-1}(i), \mathrm{BIN}_n^{-1}(j))$ pre všetky i a j, kde $M_f(i,j)$ je j-ty porvok i-teho riadku matice $M_f(i,j)$.

Pri analyzovaní takejto matice sa budeme zameriavať na oblasti zvané *kombinatorické obdĺžniky*, ďalej zvané tiež iba *obdĺžniky*.

Definícia 3.2.2. [EK97] Nech X a Y sú ľubovoľné množiny. Potom ľubovoľnú podmožinu R množiny $X \times Y$ nazývame *oblžnikom*, ak platí ,že ak $(x_1, y_1) \in R$ a $(x_2, y_2) \in R$,potom aj $(x_1, y_2) \in R$ a aj $(x_2, y_1) \in R$.

Špeciálne, nech $X = \{0,1\}^n$ a $Y = \{0,1\}^n$, $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$,$c \in \{0,1\}$, potom *c-monochromatickým obdĺžnikom* nazývame R, ak je to *obdĺžnik* a pre všetky $(x, y) \in R$ platí $M_f(\mathrm{BIN}(x), \mathrm{BIN}(y)) = c$

Ďalej uvedieme užitočnú vlastnosť súvisiacu s hore uvedenými typmi množín a protokolmi, ktoré počítajú nejakú funkciu.

Definícia 3.2.3. Nech $\mathcal{P}$ je protokol a v je vrchol jeho stromu. Označením R_v sa myslí množina vstupov (x, y), pri ktorých sa pri výpočte protokolu $\mathcal{P}$ prejde vrcholom v.

Lema 3.2.1. *[EK97] R_v je vždy obdĺžnik.*

Dôkaz. Tvrdenie platí pre koreň. Pre ostatné vrcholy dokážeme platnosť tvrdenia indukciou. Myšlienka je taká, že potomkom vrcholu v iba rozdelíme *obdĺžnik* priradený vrcholu v. □

Dôsledok. *Ak v je list, R_v je vždy monochromatický obdĺžnik.*

Dôkaz. Keďže v protokle $\mathcal{P}$ hodnota v liste je hodnota funkcie f, pre parametre, ktoré vedú k tomuto listu, musia byť všetky tieto hodnoty rovnaké. □

Z predošlej vety a dôsledku vyplýva, že každý protokol počítajúci f indukuje disjunktné pokrytie matice M_f *monochromatickými obdĺžnikmi*. Tento fakt môžeme použiť na dolný odhad komunikačnej zložitosti každej funkcie.

Lema 3.2.2. *[EK97] Ak každé disjunktné pokrytie M_f monochromatickými obdĺžnikmi potrebuje aspoň k obdĺznikov, potom $D(f) \geq \log k$*

Otázka, či dolný odhad získaný hore uvedenou lemou je vždy tesný je otvorená [EK97]. Isté je, že odhad je najviac kvadraticky zlý [EK97].

3.2.1 *Klamúce množiny*

Vypočítať veľkosť minimálneho disjunktného pokrytia M_f môže byť veľmi ťažké, ale existujú jednoduchšie metódy pre dolný odhad komunikačnej zložitosti. Jednou z nich je metóda *klamúcich množín*. Tá vlastne priamo dáva iba dolný odhad minimálneho disjunktného pokrytia, avšak tento odhad môže byť až exponenciálne zlý [Hro97].

Na použitie tejto techniky je treba nájsť veľkú množinu vstupov, takú, že žiadne dva prvky z tejto množiny nemôžu byť v jednom *monochromatickom obdĺžniku*.

Definícia 3.2.4. Nech $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. Množina $S \subset \{0,1\}^n \times \{0,1\}^n$ sa nazýva *klamúca množina*, ak existuje $z \in \{0,1\}$ také, že pre všetky $(x,y) \in S, f(x,y) = z$ a pre každé dva rôzne páry (x_1,y_1) a (x_2,y_2) buď $f(x_1,y_2) \neq z$ alebo $f(x_2,y_1) \neq z$.

Lema 3.2.3. *[EK97] Ak f má klamúcu množinu S veľkosti t, veľkosť minimového disjunktného pokrytia M_f monochromatickými obdĺžnikmi je aspoň t*

Dôsledok. *Ak f má klamúcu množinu S veľkosti t, potom $D(f) \geq \log t$*

3.3 Aplikácie komunikačnej zložitosti

Do tejto časti zaraďujem tie tvrdenia, ktoré nesúvisia iba samoúčelne s komunikačnou zložitosťou, ale poskytujú dolné odhady iných zložitostných mier v iných modeloch. Ako prvý uvedieme výsledok, ktorý sa objavil medzi prvými, v prvých článkoch pojednávajúcich o komunikačnej zložitosti. Kapitolu ukončíme aplikáciou komunikačnej zložitosti v sekvenčných výpočtoch.

3.3.1 VLSI

VLSI chip (integrovaný obvod) sa dá abstraktne popísať ako mriežka, na ktorej v každom políčku je buď procesor, alebo spojovací vodič. Technologicky sú tieto súčiastky veľmi malé, preto ich možno umiestniť veľmi veľa na malú plochu, a tým dosiahnuť vysokú mieru paralelizmu. Komunikačná zložitosť nám poskytuje metódu ako dokázať dolný odhad pre súčin plochy, ktorú chip zaberie (A) a druhú mocninu paralelného času (T), ktorý chip potrebuje na vypočítanie funkcie f.

Veta 3.3.1. *[EK97]* $D^{\mathrm{BEST}}(f) \leq \sqrt{AT}$

3.3.2 Logické obvody

Rozdiel medzi logickým obvodom a integrovaným obvodom je ten, že logický obvod je vždy acyklicky orientovaný graf, kým integrovaný obvod môže obsahovať cykly. Ak uplatníme určité obmedzenia pre rozmiestnenie logického obvodu do mriežky, platí pre minimálnu plochu A nasledovný dolný odhad:

Veta 3.3.2. *[Hro97]* $D^{\mathrm{BEST}}(f) \leq \sqrt{A}$

3.3.3 Čas a priestor pre Turingové stroje

Pomocou deterministickej komunikačnej zložitosti sme schopní nájsť dolné odhady pre súčin časovej a priestorovej zložitosti ($S(n)T(n)$) pre Turingov stroj akceptujúci daný jazyk. Avšak odhady pomocou tejto metódy sú najviac kvadratické.

Lema 3.3.3. *[EK97] Nech máme funkciu* $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *a nech* M *je viacpáskový turingov stroj so vstupnou páskou iba na čítanie, ktorý na každom vstupe dĺžky* n *urobí najviac* $T(n)$ *krokov a používa priestor* $S(n)$ *a slová tvaru* $x0^n y$*,kde* $|x| = |y|$ *akceptuje práve vtedy keď* $f(x,y) = 1$*. Potom* $D(f) = O(T(n)S(n)/n)$

Príklad 3.3.1. [EK97]

Uvažujme jazyk palindrómov, $L = \{ww^R\}$ a v ňom špeciálny prípad, keď posledná tretina slova w sú samé 0. V tomto prípade je funkcia f ekvivalentná funkcii f_{EQU}, takže $D(f) = n$. Podľa vety 3.1.2, $T(n)S(n) = \Omega(n^2)$. Ako je zrejmé, tento jazyk sa dá akceptovať v lineárnom čase, s použitím lineárnej pamäte tak, že skopírujeme prvé slovo na pracovnú pásku, a potom porovnáme. Alebo vieme tento jazyk akceptovať v kvadratickom čase s logaritmickou pamäťou, keď si budeme na pracovnej páske pamätať iba index, a budeme chodiť hore dole po vstupnej páske a postupne porovnávať prefixy slova w.

3.3.4 Čas v jednopáskovom Turingovom stroji

Komunikačná zložitosť sa dá použiť aj na získanie dolného odhadu času, ktorý potrebuje Turingov stroj na akceptovanie nejakého jazyka. Ale zatiaľ nájdený spôsob používa namiesto deterministickej komunikačnej zložitosti istú randomizovanú zlozitosť. Táto zložitostná miera je podobná ako randomizovaná miera, ktorú sme popísali v prvej časti kapitoly. Líši sa tým, že obaja účastníci majú spoločný zdroj náhodnosti. Pričom od protokolu požadujeme, aby dával dobrý výsledok s pravdepodobnosťou 1 a ako cenu protokolu berieme očakávanú dĺžku cesty pre najhorší vstup (nie teda najhoršiu cestu pre najhorší vstup). Dolné odhady, ktoré možno dosiahnuť sú až kvadratické. Nájsť dolné odhady, ktoré by boli väčšie ako lineárne, pre viacpáskové Turingove stroje sa ešte nikomu nepodarilo [Hro97].

Lema 3.3.4. *[EK97] Nech máme funkciu* $f\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *a nech* M *je jednopáskový Turingov stroj, ktorý beží v čase* $T(n)$ *a slová tvaru* $x0^n y$*, kde* $|x| = |y|$ *akceptuje práve vtedy keď* $f(x,y) = 1$*.*

Potom $R_0^{pub}(f) = O(T(n)/n)$

Príklad 3.3.2. Opäť uvažujme jazyk palindrómov. $R_0^{pub}(EQ) = \Theta(n)$[EK97], z čoho vyplýva $T(n) = \Omega(n^2)$.

3.3.5 Konečné automaty

Pre deterministické automaty je zjavné, že všetka komunikácia ktorá je dostatočná, je stav, ktorý má automat pri prechode z ľavej polovice vstupu na pravú. Tento výsledok nemá však žiaden význam, lebo poznáme efektívne metódy, ako pre deterministické konečné automaty nájsť minimálny deterministický konečný automat.

Definícia 3.3.1. Nech L je regulárny jazyk.

$s(L)$ je veľkosť najmenšieho deterministického konečného automatu, ktorý akcpetuje jazyk L.

$ns(L)$ je veľkosť najmenšieho nedeterministického konečného automatu, ktorý akceptuje jazyk L.

Jednosmerná komunikácia znamená, že v protokole správy posiela iba jeden účastník.

Definícia 3.3.2. $D_1(f)$ je komunikačná zložitosť, pri jednosmernej komunikácii

$N_1(f)$ je komunikačná zložitosť, pri jednosmernej nedeterministickej komunikácii.

Veta 3.3.5. *[Hro97] Nech* $L = \{xy \mid f(x,y) = 1\}$. *Potom* $D_1(f) \leq \lceil \log_2 s(L) \rceil$.

Pre nedeterministické automaty je už otázka zaujímavejšia, odpoveď je veľmi podobná deterministickému prípadu.

Veta 3.3.6. *[Hro97] Nech* $L = \{xy \mid f(x,y) = 1\}$. *Potom* $N_1(f) \leq \lceil \log_2 ns(L) \rceil$.

Kapitola 4

$g(n)$-nevyvážené rozdelenie

V tejto kapitole sa budeme zaoberať problémom, ktorý vznikne, ak by sme mali taký komunikačný problém, v ktorom nie je možné rozdeliť *bity* presne na polovicu. Napríklad v dôkaze vety 3.3.1 v [1] je nutné, aby sa dali vstupné *bity* rozdeliť presne na polovicu, čo pre nepárny počet portálov nie je možné, a v prípade, že počet vstupných *bitov*, ktoré vstupujú do jedného portálu je $\omega(1)$, môže viesť zanedbanie tohoto faktu k nepravdivým výsledkom. Vo zvyšku tejto kapitoly toto tvrdenie, ktoré je vlastným výsledkom, dokážeme.

Poznámka. Veta podobná k vete 3.3.1 sa vyskytuje aj v [2], kde je však hore uvedený problém irelevantný, keďže v onej vete sa namiesto vyváženého premenlivého rozdelenia používa takmer vyvážené, v ktorom môže byť rozdiel v počtoch *bitov* ktoré dostanú počítače až $2n/3$. Na druhej strane, môžu byť dolné odhady podľa tejto vety menšie ako podľa vety používajúcej vyvážené rozdelenie.

4.1 Definícia, základné poznatky

Definícia 4.1.1. $D_g^{\mathrm{B}EST}(f) = \min_{|I|=n+g(n)} D(v(f,I))$.

Nasledijúce dve vety sú zrejmé.

Tvrdenie 4.1.1. $D_g^{\mathrm{B}EST}(f) < D^{\mathrm{B}EST}(f) + g(n)$.

Tvrdenie 4.1.2. $D^{\mathrm{B}EST}(f) < D_g^{\mathrm{B}EST}(f) + g(n)$.

4.2 Vlastné výsledky

Veta 4.2.1. *Pre každú funkciu* $g\colon \{0,1,2\ldots n-1,n\} \to \{0,1,2,\ldots n-1,n\}$, *,existuje taká funkcia* $f\colon \{0,1\}^{2n} \to \{0,1\}$, *že* $D_g^{\mathrm{BEST}}(f) \leq 1$ *a* $D^{\mathrm{BEST}}(f) = g(n)$.

Dôkaz. Nech $f' \colon \{0,1\}^{n+g(n)} \to \{0,1\}$a $D^{\mathrm{BEST}}(f') = (n+g(n))/2$. Podľa vety 3.1.6 vieme, že takáto funkcia existuje. Nech $f \colon \{0,1\}^{2n} \to \{0,1\}$ je vytvorená pomocou f', takým spôsobom, že iba pridáme premenné (bezvýznamné bity) na ktorých funkčná hodnota f nezávisí[1].

Zjavne $D_g^{\mathrm{BEST}}(f) \leq 1$, lebo jeden počítač si môže zobrať až $n + g(n)$ *bitov*, takže funkciu f'a tým pádom aj f dokáže vypočítať triviálne. Rovnako triviálny je aj dôkaz $D^{\mathrm{BEST}}(f) \leq g(n)$, stačí zobrať také rozdelenie, v ktorom prvý počítač dostane n významných *bitov*, a druhý počítač mu zvyšných $g(n)$ významných *bitov* pošle. Dôkaz obrátenej nerovnosti urobíme sporom. Predpokladajme, že existuje také rozloženie *bitov* I, v ktorom prvý počítač dostane $b \leq n$ významných vstupných *bitov* (bez ujmy na všeobecnosti môžeme predpokladať, že prvý počítač nedostane menej významných *bitov* ako druhý) a protokol $\mathcal{P}$ počítajúci f ktorý potrebuje $k < g(n)$ *bitov*. Potom ale vieme spraviť protokol $\mathcal{P}'$, ktorý bude počítať funkciu f' s komunikáciou dĺžky

$$b - (n + g(n))/2 + k < n - (n + g(n))/2 + g(n) = (n + g(n))/2 \qquad (4.1)$$

Čo je spor s predpokladom, že $D^{\mathrm{BEST}}(f') = (n + g(n))/2$. Rozloženie I' pre protokol $\mathcal{P}'$ bude také, že prvý počítač bude mať ľubovoľných $\frac{n+g(n)}{2}$ bitov, ktoré dostal prvý počítač v rozdelení I. V protokole $\mathcal{P}'$ pošle najprv druhý počítač hodnoty *bitov* z $I \setminus I'$, tých je najviac $b - (n + g(n))/2$. Ďalej bude pracovať ako protokol $\mathcal{P}$, ktorý posiela k *bitov*. □

Poznámka. Veta platí aj v okrajových prípadoch pre funkciu g, avšak treba vhodne zadefinovať degenerovaný prípad, keď jeden počítač pozná celý vstup, a druhý počítač nemá žiaden vstup.

K predchádzajúcej vete existuje aj symetrická veta, ktorej dôkaz je tiež vlastný výsledok.

Veta 4.2.2. *Pre každú funkciu* $g \colon \{0,1,2 \ldots n-1,n\} \to \{0,1,2,\ldots n-1,n\}$, *existuje taká funkcia* $f \colon \{0,1\}^{2n} \to \{0,1\}$, *že* $D^{\mathrm{BEST}}(f) \leq 1$ *a* $g(n) \leq D_g^{\mathrm{BEST}}(f) \leq g(n) + 1$.

Dôkaz. Nech $f' \colon \{0,1\}^n \to \{0,1\}$ je taká, že $D^{\mathrm{BEST}}(f') = n/2$. Zostrojíme funkciu $f \colon \{0,1\}^{2n} \to \{0,1\}$, takú že $f(xy) = f'(x) \wedge f'(y)$. Zjavne $D^{\mathrm{BEST}}(f) \leq 1$ pretože stačí ak si prvý počítač vezme prvých n *bitov*, a druhému počítaču pošle výsledok $f'(x)$, ktorý vie vypočítať a druhý počítač vie už potom vypočítať $f'(x) \wedge f'(y)$. Ďalej tiež zjavne $D_g^{\mathrm{BEST}}(f) \leq g(n) + 1$, pretože stačí ak si prvý počítač zoberie celé x a $g(n)$ *bitov* z y a druhému

[1] $f(x_0x_1x_2 \ldots x_{2n-1}) = f'(x_0x_1x_2 \ldots x_{n+g(n)-1})$

počítaču pošle $f'(x)$ a $g(n)$ *bitov* z y, ktoré mal na vstupe. Ten už vie vypočítať $f(xy) = f'(x) \wedge f'(y)$. Dôkaz opačnej nerovnosti urobíme sporom, predpokladajme, že existuje rozdelenie I a protokol $\mathcal{P}$ počítajúci f, ktorému stači poslať $k < g(n)$ bitov. Predpokladajme bez ujmy na všeobecnosti, že v rozložení I, dostane druhý počítač (ten ktorý má vstup $n - g(n)$ *bitov*) a *bitov* z x, a b *bitov* z y a $a \leq b$. Potom ale dokážeme nájsť rozloženie I' a protokol P', ktorý by počítal f' s komunikáciou dĺžky

$$b - n/2 + k < n - g(n) - n/2 + g(n) = n/2 \tag{4.2}$$

Čo je v rozpore s tým, že $D^{\mathrm{BEST}}(f') = n/2$. Rozloženie I' vyzerá tak, že druhý počítač dostane na vstup ľubovoľných $n/2$ *bitov* ktoré mal druhý počítač a boli z y. Protokol $\mathcal{P}'$ funguje tak, že najprv prvý počítač pošle druhému zvyšných $b - n/2$ *bitov*, ktoré sú z y a mal ich druhý počítač v I. Nasledovne sa $\mathcal{P}'$ správa ako $\mathcal{P}$ na vstupe kde je x nejaké pevne dané a $f'(x) = 1$. □

Kapitola 5

Zložitosť *2-CNF* funkcií

V tejto kapitole sa budeme zaoberať funkciami, ktoré majú istým spôsobom obmedzenú deskriptívnu zložitosť. V [Hro97] sa tvrdí, že doteraz nebola nájdená[1] funkcia, na ktorej zostrojenie by bolo treba viac ako lineárne veľký počet logických hradiel. A ďalej sa tvrdí, že takmer všetky funkcie, ktorých *komunikačná zložitosť* je lineárne veľká, môžu byť kandidátom na takúto funkciu. My ukážeme, že existuje funkcia, ktorá má lineárne veľkú *komunikačnú zložitosť*, avšak na jej konštrukciu stačí lineárne veľa hradiel, ba čo viac, táto konštrukcia je aj v špeciálnom tvare. Tento výsledok sme získali popri skúmaní komunikačnej zložitosti funkcií, ktoré sa dajú reprezentovať *2-CNF* formulov.

5.1 Reprezentácia funkcií

Každú funkciu možno reprezentovať jednoznačne vektorom jej hodnôt. Ak funkcia zobrazuje množinu $\{0,1\}^n$ do $\{0,1\}$, má tento zápis dĺžku 2^n. Z výpočtového hľadiska sú vhodnejšie nasledovné reprezentácie, z nich pre túto kapitolu je najdôležitejšia reprezentácia formulov v *konjunktívnom normálnom tvare*.

5.1.1 Logické obvody

Logický obvod je zariadenie, ktoré sa dá použiť na paralelné vypočítanie nejakej funkcie. Pozostáva z hradiel počítajúcich elementárne logické funkcie a liniek, ktoré prenášajú výsledky medzi jednotlivými hradlami. Formálne možno určitú podtriedu[2] logických obvodov n vstupmi popísať nasledovne:

[1] Tým sa myslí, že pre žiadnu konkrétnu funkciu nebolo dokázané.

[2] trieda sa nazýva 2 fan-in

Definícia 5.1.1. [Hro97] Logický obvod je acyklický orientovaný graf v ktorom:

1. Existuje práve n vstupných vrcholov. Do týchto vrcholov nevchádza žiadna hrana.
2. Každý vrchol, do ktorého vchádza aspoň jedna hrana, sa nazýva hradlo. Ku každému hradlu je priradená nejaká elementárna logická funkcia.
3. Existuje práve 1 výstupný vrchol. Z neho nevychádza žiadna hrana.

Výpočet takéhoto logického obvodu prebieha prirodzene, zo vstupných vrcholov sa prenesú hodnoty do hradiel, s ktorými sú spojené linkami hodnoty, hradlá vypočítaju svoju funkciu a výsledok pošlú cez všetky odchádzajúce hrany ostatným hradlám, a tak ďalej. Výsledok je potom uložený v jedinom výstupnom vrchole.

5.1.2 Logické formuly, *CNF*

Logická formula je dobre uzátvorkovaný výraz, ktorý používa symboly pre premenné, binárne a unárne operácie. Pre asociatívne a zároveň komutatívne operácie možno zátvorky vynechávať, symboly pre premenné sú väčšinou x_i alebo y_i. Logická formula s premennými $\{x_0, x_1, x_2 \dots x_{n-1}\}$, reprezentuje tú funkciu $f\colon \{0,1\}^n \to \{0,1\}$, ktorej funkčnú hodnotu pre $f(w)$ dostaneme po substituovaní $w(i)$ za x_i , pre všetky i, vo formuli.

Definícia 5.1.2. *Formula je v konjunktívnom normálnom tvare s k premennými na klauzulu* (skrátene *k-CNF*), ak sa dá napísať ako $\bigwedge_{i\leq m} \bigvee_{j<k} Q_{ij}$, kde Q_{ij} je x_{ij} alebo $\neg x_{ij}$.

Príklad 5.1.1. Formula $(x_1 \vee x_2) \wedge (x_3 \vee \neg x_2) \wedge (x_1 \vee \neg x_4) \wedge (\neg x_1 \vee x_3) \wedge (x_4 \vee \neg x_3)$ je v *konjunktívnom normálnom tvare s 2 premennými na klauzulu.*

Keď je formula v *konjunktívnom normálnom tvare*, hovoríme, že je konjunkciou disjunktívnych *klauzúl.*

Poznámka. 2-CNF funkcie budeme volať tie funkcie, ktoré sa dajú reprezentovať *2-CNF* formulou.

Poznámka. Ak vynecháme z definície *k-CNF* požiadavku, že všetky klauzule majú práve k prvkov, dostaneme iba *konjunktívnu normálnu formu (CNF)*

Tvrdenie 5.1.1. *Každú funkciu $f\colon \{0,1\}^n \to \{0,1\}$ možno reprezentovať logickou formulou v špeciálnom tvare, zvanom konjunktívna normálna forma, skrátene CNF.*

Dôkaz. Pre každú hodnotu parametra $w = w_0w_1w_2 \dots w_{n-1}$, kde $f(w) = 0$. Dáme do formuly klauzulu $(g(w_0, x_0) \vee g(w_1, x_1) \vee g(w_2, x_2) \vee \cdots \vee g(w_{n-1}, x_{n-1}))$, kde $g(w_i, x_i)$ je x_i ak w_i je 0, alebo $\neg x_i$ ak w_i je 1. □

Tvrdenie 5.1.2. *Existuje funkcia, $f\colon \{0,1\}^n \to \{0,1\}$, ktorá sa nedá reprezentovať 2-CNF formulou.*

Dôkaz. *2-CNF* formúl je $2^{O(n^2)}$, kým všetkých funkcií je presne 2^{2^n}. □

Ďalej budeme pracovať s *2-CNF* formulami, ktoré v klauzulách nepoužívajú negáciu. Preto budeme môcť hovoriť o izomorfizme formúl a označených grafov. Tento izomorfizmus získame tak, že vrcholy grafu, na ktorý formulu zobrazíme, budú označené symbolmi pre premenné, a klauzule zobrazíme na hrany medzi dvoma symbolmi, ktoré sa v klauzuli nachádzajú. Takýto prístup je vhodný, lebo neskôr sa ukáže, že problém, ktorým sa zaoberáme je vlastne aj tak iba grafovým problémom.

5.2 *2-CNF* a *3-CNF*

Dôvod, prečo sme skúmali *2-CNF* funkcie je v tom, že problém zistiť, či boolovská formula zapísaná v *3-CNF* je splniteľná je **NP**-úplný, avšak pre *2-CNF* formule existuje jednoduchý algoritmus [AB79], ktorý zistí, či je formula splniteľná.

Spomínaný algoritmus sa dá popísať nasledovne: Každú premennú a jej negáciu reprezentujeme vrcholom v grafe, a každú klauzulu $(a \vee b)$ nahradíme dvojicou orientovaných hrán $(\neg a, b)$ a $(\neg b, a)$. Formula je splniteľná práve vtedy, ak v grafe neexistuje kružnica, na ktorej leží súčasne premenná a aj jej negácia.

V záujme zistiť, či sa jednoduchosť rozhodovania splniteľnosti *2-CNF* funkcií prejaví aj v komunikačnej zložitosti, sa budeme zaoberať komunikačnou zložitosťou *2-CNF funkcii*.

Ak by sme skúmali iba komunikačnú zložitosť s pevným rozdelením, prišli by sme na to, že existuje jednoduchá funkcia vyjadriteľná v *2-CNF* , ktorá má lineárnu komunikačnú zložitosť. Je to napríklad funkcia f_{EQU}, definovaná v kapitole 3. Preto sa budeme zaoberať s *vyváženým premenlivým rozdelením vstupu*.

Pre *3-CNF* je otázka vyriešená, lebo negácia problému, či graf obsahuje trojuholník, sa da zapísať v *3-CNF* a je dokázané, že problém, či graf obsahuje trojuholnik, má lineárnu *komunikačnú zložitosť* aj pri *vyváženom premenlivom rozdelení vstupu* [Hro97].

Pre *2-CNF* doteraz žiadna lineárne ťažká funkcia nebola nájdená, ani nie je dokázané, že by pre každú *2-CNF funkciu* existoval efektívnejší protokol.

Vo zvyšku tejto časti dokážeme, že existuje lineárne ťažká[3] *2-CNF funkcia*, a že ak sa obmedzíme na *2-CNF* formuly, ktoré obsahujú každú premennú najviac dva krát, tak dostaneme triedu funkcií ktore maju komunikačnú zložitosť $O(1)$. Na dodatok uvedieme konkrétnu *2-CNF funkciu*, ktorá má komunikačnú zložitosť $\Theta(\sqrt{n})$, pretože podobná idea dôkazu je použitá v dôkaze existencie lineárne ťažkej funkcie.

5.2.1 Dolný odhad zložitosti *2-CNF* funkcií

Konkrétna funkcia

V tejto časti uvedieme dôkaz jednoduchšieho tvrdenia ako v nasledujúcej časti, avšak detailnejšie a formálnejšie predvedieme použité techniky. Zatiaľ najťažšia nájdená konkrétna funkcia reprezentovatelná *2-CNF* formulou je, že či graf neobsahuje žiadnu cestu dĺžky dva. Problém sa dá vysloviť aj ako požiadavka zistiť, či hranový graf obsahuje hranu.

Definícia 5.2.1. $L_{\text{dva}} = \{G | G \text{ obsahuje cestu dĺžky } 2\}$.

Veta 5.2.1. L^C_{dva} *je 2-CNF funkcia*

Dôkaz. Uvažujme pre každé n, že máme kompletný graf. Zostrojíme formulu tak, že pre každé dve susedné hrany A, B pridáme jednu disjunkciu $(\neg A \vee \neg B)$. Formula je splnená práve vtedy, keď z každej susednej dvojice hrán aspoň jedna chýba, teda práve vtedy keď graf neobsahuje cestu dĺžky dva. □

Poznámka. Formula, ktorú vytvoríme pre každé n je izomorfná s hranovým grafom kompletného grafu K_n.

Avšak deterministická komunikačná zložitosť je rovnaká pre problém a jeho negáciu[4]. Takže môžeme hovoriť aj o probléme či graf obsahuje cestu dĺžky dva, čo je negácia vyššie popísanej formule.

Pre dôkaz vety 5.2.6, musíme uvážiť všetky rôzne rozdelenia vstupu. To znamená, že nevieme, ktorý účastník má na začiatku informácie o ktorých hranách. K tomuto problému pristúpime tak, že budeme skúmať hranové ofarbenia úplného grafu červenou a modrou farbou tak, aby bol rovnaký počet modrých a červených vrcholov. Čo bude zodpovedať tomu prípadu, keď účastník A dostane modré hrany a účastník B dostane červené hrany. Budeme sa zaoberať vstupmi, kde počet vrcholov grafu je deliteľný štyrmi. Potom počet hrán grafu je párny a takýto graf nazývame ofarbený graf.

[3] V zmysle komunikačnej zložitosti, tak ako bolo uvedené vyššie

[4] Je zrejmé, že stačí zameniť hodnoty v listoch protokolového stromu

Definícia 5.2.2. Ofarbený graf je usporiadaná dvojica (G, f), kde $G = (V, E)$ je graf, a $f : E \to \{0, 1\}$ je funkcia udávajúca farbu hrany, ak $f(e) = 0$ hovoríme, že hrana e je modrá a ak $f(e) = 1$ hovoríme, že hrana e je červená. Navyše musí platiť $|\{e|f(e) = 1\}| = |\{e|f(e) = 0\}|$.

Teraz vyslovíme zopár tvrdení o ofarbených grafoch, pomocou ktorých budeme môcť dokázať existenciu dostatočne veľkej *klamúcej množiny*, pre každé rozdelenie vstupu I.

Lema 5.2.2. *Nech G je kompletný ofarbený graf a X je množina vrcholov. Nech graf $G \setminus X$ obsahuje aspoň jednu červenú hranu a aspoň jednu modrú hranu, potom graf $G \setminus X$ obsahuje aspoň jeden vrchol incidentný s modrou aj červenou hranou.*

Dôkaz. Dôkaz je ľahký, ak by neexistoval taký vrchol, potom by musel mať graf $G \setminus X$ aspoň dva komponenty, jeden, kde sú iba vrcholy incidentné s červenými hranami a druhý, kde sú vrcholy incidentné s modrými hranami, ale to nemôže byť, lebo G je $n - 1$ vrcholovo súvislý. □

Lema 5.2.3. $\binom{n}{2}/2 > \lfloor n/8 \rfloor (n - 1)$ *pre všetky $n = 8k$ väčšie ako* 1.

Dôkaz.

$$\begin{aligned} \binom{n}{2}/2 - \lfloor n/8 \rfloor (n-1) &= 2k(8k-1) - k(8k-1) = \\ &= 16k^2 - 2k - 8k^2 + k = \\ &= 8k^2 - k = k(8k-1) > 0 \end{aligned} \tag{5.1}$$

□

Tvrdenie 5.2.4. *Nech G je kompletný ofarbený graf s n vrcholmi, potom obsahuje aspoň $\lfloor n/24 \rfloor$ trojíc rôznych vrcholov (a_i, b_i, c_i), $1 \leq i \leq n/24$, takých že:*

$$\begin{aligned} &f(a_i b_i) = 0 \wedge f(a_i c_i) = 1 \quad \textit{pre všetky } i \\ &\{a_i, b_i, c_i\} \cap \{a_j, b_j, c_j\} = \emptyset \quad \textit{pre všetky } i, j \textit{ také, že } (i \neq j) \end{aligned} \tag{5.2}$$

Dôkaz. Zrejme jednu takúto trojicu vieme vďaka vete 5.2.2 vybrať. Ďalej postupujeme indukciou od počtu vybraných trojíc. Sporom dokážeme platnosť tvrdenia, nech sa dá vybrať $k < \lfloor n/24 \rfloor$ trojíc spĺňajúcich 5.2, nech $X = \bigcup_{1 \leq i \leq k} \{a_i, b_i, c_i\}$ potom graf $G \setminus X$ bude obsahovať aspoň $\binom{n}{2}/2 - (3k)(n-1) > \binom{n}{2}/2 - \lfloor n/8 \rfloor (n-1) > 0$ (5.2.3) červených aj modrých hrán, podľa lemy 5.2.2 teda možno vybrať aj $k + 1$ trojíc vrcholov. □

Aby sme uľahčili dôkaz nasledujúcej vety, zadefinujeme nasledujúce pojmy, ktoré súvisia s predchádzajúcou vetou. Nech T je systém trojíc (a_i, b_i, c_i) spĺňujúcich 5.2

first(i) je pozícia *bitu*, ktorý je priradený hrane $a_i b_i$, niekedy, keď je z kontextu zrejmé, pod týmto výrazom myslíme priamo hranu $a_i b_i$,

second(i) je pozícia *bitu*, ktorý je priradený hrane $a_i c_i$, niekedy, keď je z kontextu zrejmé, pod týmto výrazom myslíme priamo hranu $a_i c_i$,

bezvýznamný bit , je *bit* s takou pozíciou, že sa nevyskytuje ako hodnota $first(i)$ ani $second(i)$ pre žiadne i,

bezvýznamná hrana , je taká hrana, ktorá je reprezentovaná bezvýznamným *bitom*,

významná hrana , je taká hrana, ktorá je reprezentovaná *bitom*, ktorý nieje bezvýznamný.

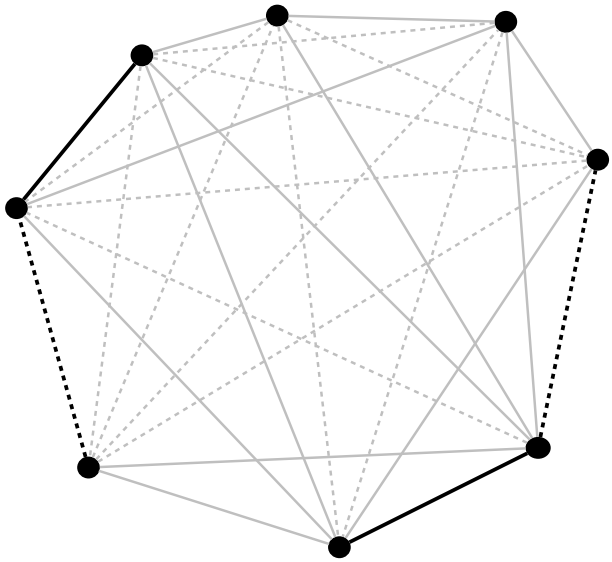
Tvrdenie 5.2.5. *Dve významné hrany sú incidentné práve vtedy, ak existuje i také, že jedna je $first(i)$ a druhá $second(i)$*

Dôkaz. Jedna implikácia je zrejmá, lebo $first(i)$ je hrana (a_i, b_i) a $second(i)$ je hrana (b_i, c_i), tieto hrany sú incidentné. Obrátenú implikáciu dokážeme nasledovne, nech pre $i \neq j$ by boli nejaké dve hrany incidentné, tak musia mať grafy nimi indukované spoločný aspoň jeden vrchol. Ale podľa (5.2) je množina $\{a_i, b_i, c_i\}$ disjunktná s $\{a_j, b_j, c_j\}$, čo je spor. □

Veta 5.2.6. $\mathrm{D}^{\mathrm{BEST}}(L_{\mathrm{dva}}) = \Theta(\sqrt{m})$. *Kde m je dĺžka vstupu, čo je zároveň počet hrán K_n a to je $\binom{n}{2}$.*

Dôkaz. Najprv dokážeme dolný odhad.

Nech $k = \lfloor n/24 \rfloor$. Keďže dokazujeme dolný odhad $\mathrm{D}^{\mathrm{BEST}}(L_{\mathrm{dva}})$, musíme dokázať dolný odhad pre všetky možné rozdelenia vstupu. Urobíme to tak, že zostrojíme *klamúcu množinu* S o veľkosti 2^k, pre ľubovolné rozloženie vstupných bitov. Ďalej teda vo zvyšku dôkazu predpokladáme, že máme dané jedno pevné rozdelenie vstupu. To rozdelenie je dané zafarbením grafu G, červené hrany dostane účastník B, a modré účastník A. Nech T je systém trojíc vrcholov z ofarbeného grafu G, $T = \bigcup_{i=1}^{k} (a_i, b_i, c_i)$ splňujúcich 5.2, podľa vety 5.2.4 môžeme predpokladať, že $k \geq \lfloor n/24 \rfloor$. Do množiny S vyberieme také vstupy w, že všetky bezvýznamné bity majú hodnotu 0 a pre $1 \leq i \leq k$ platí $w_{first(i)} \neq w_{second(i)}$. Takýchto vstupov je 2^k, lebo pri vytváraní urobíme k nezávislých rozhodnutí a to, že či $w_{first(i)} = 1$ a $w_{second(i)} = 0$ alebo $w_{first(i)} = 0$

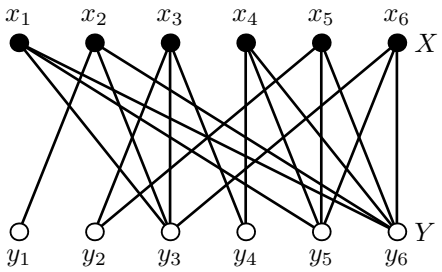


Obr. 5.1: Tu vidíme príklad ofarbenia grafu a výber dvoch trojíc.

a $w_{second(i)} = 1$. Ostáva dokázať, že S je *klamúca množina*. Najprv ukážeme, že pre všetky $w \in S$ je $w \notin L_{\text{dva}}$. Keďže všetky bezvýznamné *bity* majú hodnotu nula, každá cesta dĺžky dva musí byť tvorená jedine významnými hranami. Podľa vety 5.2.5 sú tieto dve hrany unikátne pre každé $1 \leq i \leq k$, a sú to hrany $first(i)$ a $second(i)$, tieto *bity* však pre žiadne slovo $w \in S$ nie sú rovnaké, takže nemôžu byť oba 1. Teraz potrebujeme dokázať, že keď zoberieme 2 rôzne slová $u \in S$,$v \in S$ a zameníme ich polovice, tak dostaneme slovo $w \in L_{\text{dva}}$. Označme teraz $u = u_1u_2$ a $v = v_1v_2$ je rozdelenie slov u a v na dve polovice, z ktorej každú dostane jeden počítač. Nech $w1 = u_1v_2$ a $w2 = v_1u_2$. Ak $u \neq v$, potom aj $u_1 \neq v_1$ a $u_2 \neq v_2$. Musí existovať také i, že $w_1(first(i)) = w_1(second(i))$, ak $w_1(first(i)) = 0$ potom $w_2(second(i)) = 1$ a $w_2(first(i)) = 1$. Takže $w1 \in L_{\text{dva}}$ alebo $w2 \in L_{\text{dva}}$. Podľa dôsledku lemy 3.2.3 $\text{D}^{\text{BEST}}(L_{\text{dva}}) = \Omega(\sqrt{m})$. Zrejme $\text{D}^{\text{BEST}}(L_{\text{dva}}) = O(\sqrt{m})$. Lebo stačí ak prvý hráč pošle n bitov o tom, s ktorými vrcholmi má incidentnú nejakú hranu. □

Existencia lineárne ťažkej funkcie

Celá táto časť je dôkazom existencie lineárne ťažkej *2-CNF funkcie*. Dôkaz je založený na istej kombinatorickej úvahe. Ukážeme, že existuje taká formula s $2n$ premennými, že pre každé rozdelenie vstupu bude existovať *klamúca množina* o veľkosti $2^{\Omega(n)}$. Problém hľadania tejto formuly môžeme pomocou



Obr. 5.2: Príklad znázornenia formuly ako grafu. Formula je uvedená v rovnici (5.4)

izomorfizmu popísanom na začiatku kapitoly vysloviť ako problém hľadania istého grafu $G = (V, E)$.

Graf, ktorý hľadáme, musí mať pre každú množinu I, ktorá obsahuje polovicu vrcholov, dosť veľa hrán, ktoré sú v $(I, V \setminus I)$-reze. Pretože inak by sme mohli jednoducho rozdeliť vstup medzi účastníkov podľa I a oni by si iba vymenili toľko bitov, koľko je hrán v tom reze. Ak však vezmeme úplný graf, dostaneme natoľko triviálnu funkciu, že má najviac logaritmickú komunikačnú zložitosť. Podobné sa stalo vždy, keď sme skúšali nejaký jednoduchý pravidelný graf. Preto sme sa rozhodli použiť pravdepodobnostnú metódu, náhodne vygenerujeme bipartitný graf s partíciami X, Y a ľavým stupňom c a dokážeme, že s veľkou pravdepodobnosťou vygenerujeme taký, ktorý bude mať pre každý rez, ktorý je medzi množinami obsahujúcimi polovicu vrcholov, aspoň $\lambda|V|/2$ vrcholov incidentných s rezovými hranami. Navyše tieto vrcholy budú z jednej partície a na jednej strane rezu. Vďaka tejto vlastnosti spolu s bipartitnosťou a konštantnosťou stupňov ľavej partície, budeme už môcť existenciu dostatočne veľkej *klamúcej množiny* dokázať pre každé rozdelenie.

Počet všetkých označených bipartitných grafov s partíciami veľkostí n a n a ľavým stupňom c je

$$\binom{n}{c}^n \tag{5.3}$$

Formula ktorú hľadáme môže vyzerať aj takto:

$$\begin{aligned}&(x_1 \vee y_3) \wedge (x_1 \vee y_5) \wedge (x_1 \vee y_6) \wedge (x_2 \vee y_1) \wedge (x_2 \vee y_3) \wedge \\ &\wedge (x_2 \vee y_6) \wedge (x_3 \vee y_2) \wedge (x_3 \vee y_3) \wedge (x_3 \vee y_4) \wedge (x_4 \vee y_4) \wedge \\ &\wedge (x_4 \vee y_5) \wedge (x_4 \vee y_6) \wedge (x_5 \vee y_2) \wedge (x_5 \vee y_5) \wedge (x_5 \vee y_6) \wedge \\ &\wedge (x_6 \vee y_2) \wedge (x_6 \vee y_5) \wedge (x_6 \vee y_6)\end{aligned} \tag{5.4}$$

Ďalej sa budeme snažiť zhora odhadnúť počet bipartitných grafov s partíciami X, Y a ľavým stupňom c, pre ktoré existuje malý rez, a porovnať tento počet s počtom všetkých bipartitných grafov s partíciami X, Y a ľavým stupňom c.

Pre ten účel generujme všetky podmnožiny s $n = |V|/2$ prvkami takým spôsobom, že vyberieme k vrcholov z X a zvyšných $n-k$ vrcholov z Y. Viď obrázok 5.3

Definícia 5.2.3. $P(n, k, \lambda, c)$ je počet bipartitných grafov s partíciami veľkosti n, n a ľavým stupňom c, ktoré v nejakom reze, ktorý obsahuje na jednej strane k vrcholov z X a $n-k$ z Y, majú menej ako λn vrcholov z druhej partície incidentných s rezovými hranami.

Veta 5.2.7.

$$P(n,k,\lambda,c) \leq \binom{n}{k}^2 \binom{k}{\min(\lambda n, k)} \binom{n-k}{\min(\lambda n, n-k)} \binom{\lambda n + n - k}{c}^k \binom{\lambda n + k}{c}^{n-k}$$

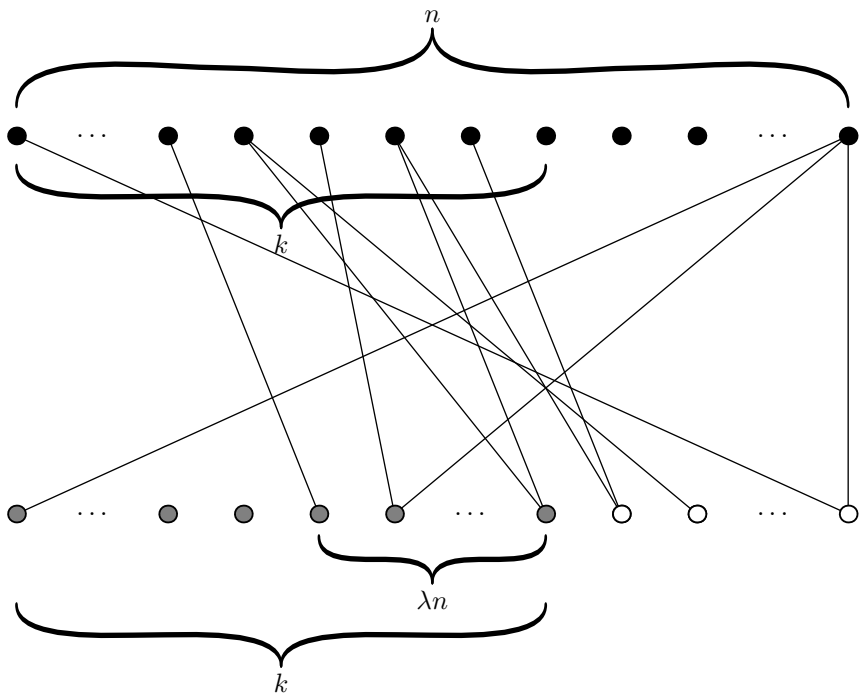
Dôkaz. Teraz vysvetlíme, prečo uvedený výraz zahŕňa všetky také grafy. Najprv vyberieme $\binom{n}{k}$ spôsobmi k vrcholov z X a $\binom{n}{n-k} = \binom{n}{k}$ spôsobmi $n-k$ vrcholov z Y a tým definujeme rez. Potom pre každú stranu rezu vyberieme λn vrcholov z druhej strany rezu, medzi ktorými môžu ísť nejaké hrany, čo je kôli bipartitnosti $\binom{k}{\lambda n}\binom{n-k}{\lambda n}$. Ak by bolo k menšie ako λn, znamená to, že z vrcholov vybraných z X by mohli ísť rezové hrany do najviac k vrcholov, čo je ale tiež menej ako λn. Preto stačí tento okrajový prípad ošetriť použitím $\min(\lambda n, k)$ v binomickom koeficiente. Rovnako ošetríme aj symetrický prípad. Nakoniec uvážime všetky možné výbery hrán tak, aby rezové hrany išli iba do vybraných vrcholov $\min(\lambda n, k)$. To je najviac $\binom{\lambda n + n - k}{c}^k$ možností a najviac $\binom{\lambda n + k}{c}^{n-k}$ možností pre symetrický prípad. Viď obrázok 5.4. □

Zosumovaním a nahradením binomických koeficientov pomocou nerovnice $\binom{n}{k} \leq 2^n$ dostaneme väčší odhad

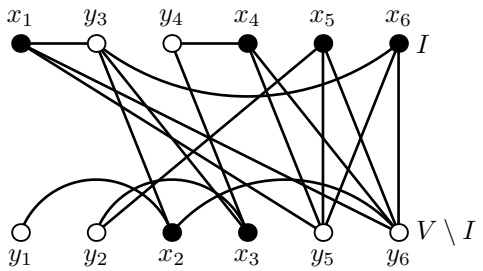
$$\sum_k P(n,k,\lambda,c) \leq \sum_k 2^{4n} \binom{\lambda n + n - k}{c}^k \binom{\lambda n + k}{c}^{n-k}$$

Oba binomické koeficienty sú vlastne polynómy stupňa c a pre dostatočne veľké n môžeme použiť, že $\binom{x}{c}$ je menšie ako $x^c/c!$. Dostaneme:

$$\sum_k P(n,k,\lambda,c) \leq 2^{4n} n c!^{-n} (\lambda n + n - k_{max})^{c k_{max}} (\lambda n + k_{max})^{c(n-k_{max})}$$



Obr. 5.3: Znázornenie k vete 5.2.7, každý z vybraných vrcholov má hranu iba s tými, ktoré sú vybrané v druhej polovici, alebo do fixnej λn prvkovej množiny z tých vrcholov, ktoré nie sú vybrané.



Obr. 5.4: Znázornenie formuly ako grafu, pri inom rozdelení na polovice. Vidno, že niektoré hrany, ktoré boli pôvodne v reze, teraz v reze nie sú.

Prekvapivo, ako dokážeme neskôr vo vete 5.2.8, $k_{max} = n/2$ je hodnota, kedy je súčin najväčší, ak budeme predpokladať, že n je párne, tak po dosadení $n/2$ za k_{max} a upravení dostaneme:

$$\sum_k P(n,k,\lambda,c) \leq 2^{5n} c!^{-n} (\lambda n + n/2)^{cn} \tag{5.5}$$

Odhad z (5.5) porovnáme s počtom všetkých uvažovaných grafov (5.3). Dostaneme nerovnicu (5.6). Potrebujeme dokázať, že existuje také c, že (5.6) platí pre nekonečne veľa n.

$$2^{5n} c!^{-n} (\lambda n + n/2)^{cn} \leq \binom{n}{c}^n \tag{5.6}$$

Pre dostatočne veľké n, $\binom{n}{c} \geq \frac{n^c}{2(c!)}$, keď tento odhad použijeme v 5.6 dostanem, že stačí dokázať:

$$2^{5n} c!^{-n} (\lambda n + n/2)^{cn} \leq 2^{-n} c!^{-n} n^c n$$

Keďže parameter λ si môžeme voliť ľubovolne z otvoreného intervalu $(0,1)$, môžme predpokladať, že $\lambda \leq 1/4$.

$$\begin{array}{rcl} 2^{5n}(3n/4)^{cn} & \leq & 2^{-n} n^{cn} \\ 2^{5n}(3/4)^{cn} n^{cn} & \leq & 2^{-n} n^{cn} \\ 2^{6n}(\frac{3}{4})^{cn} & \leq & 1 \\ 2^{6n} & \leq & (\frac{4}{3})^{cn} \\ 2^6 & \leq & (\frac{4}{3})^c \end{array} \tag{5.7}$$

Ak zvolíme c aspoň $\ln_{4/3} 64$, čo je menej ako 15, potom nerovnica bude platiť.

Ostáva nám ešte dokázať nasledovné tvrdenie, ktoré sme v odvodení použili.

Veta 5.2.8. *Výraz* $(\lambda n + n - k)^{ck} (\lambda n + k)^{c(n-k)}$ *nadobúda najväčšiu hodnotu pre* $k = n/2$.

Dôkaz. Z celého výrazu urobíme nc-tú odmocninu, urobíme substitúciu $\alpha = k/n$, vyberieme pred zátvorku n^α a $n^{1-\alpha}$, čo sú operácie, ktoré zachovávajú nerovnosti. Dostaneme $n(\lambda + 1 - \alpha)^{n\alpha} (\lambda + \alpha)^{n(1-\alpha)}$, môžeme ešte vydeliť n.

$$f(\alpha) = (\lambda + 1 - \alpha)^\alpha (\lambda + \alpha)^{1-\alpha}$$

Funkciu f zderivujeme podľa α.

$$\begin{array}{l} f'(\alpha) = (\lambda+\alpha)^{1-\alpha}(\frac{1-\alpha}{\lambda+\alpha} - \ln(\lambda+\alpha))(\lambda+1-\alpha)^\alpha + \\ +(\lambda+\alpha)^{1-\alpha}(\lambda+1-\alpha)^\alpha(\ln(\lambda+1-\alpha) - \frac{\alpha}{\lambda+1-\alpha}) \end{array} \tag{5.8}$$

Vyberieme pred zátvorku výraz $(\lambda+\alpha)^{1-\alpha}(\lambda+1-\alpha)^{\alpha}$ a dostaneme

$$f'(\alpha) = (\lambda+\alpha)^{1-\alpha}(\lambda+1-\alpha)^{\alpha}(\frac{1-\alpha}{\lambda+\alpha} - \frac{\alpha}{\lambda+1-\alpha} + \ln(\lambda+1-\alpha) - \ln(\lambda+\alpha)) \tag{5.9}$$

Ďalej budeme skúmať vlastnosti funkcie f' na otvorenom intervale $(0,1)$, bude nás zaujímať fakt, že jediný nulový bod funkcie f' je $\frac{1}{2}$. Nulovosť funkcie f', keďže $\lambda > 0$ závisí iba od výrazu $\frac{1-\alpha}{\lambda+\alpha} - \frac{\alpha}{\lambda+1-\alpha} + \ln(\lambda+1-\alpha) - \ln(\lambda+\alpha)$, označme $F(\alpha) = (\frac{1-\alpha}{\lambda+\alpha} - \frac{\alpha}{\lambda+1-\alpha} + \ln(\lambda+1-\alpha) - \ln(\lambda+\alpha))$. Ľahko vidno, že $F(\alpha) = G(\alpha) - G(1-\alpha)$, kde $G = \frac{1-\alpha}{\lambda+\alpha} + \ln(\lambda+1-\alpha)$. Z toho môžeme odvodiť:

- G je klesajúca funkcia.
- $\bar{G}(\alpha) = G(1-\alpha)$ je rastúca funkcia.
- F je klesajúca, lebo je rozdielom klesajúcej a rastúcej funkcie.
- $F(\alpha) = -F(1-\alpha)$.
- $F(\frac{1}{2}) = 0$.

Z čoho môžeme odvodiť, že jediný stacionárny bod funkcie f je $\frac{1}{2}$. □

Z predchádzajúcich úvah vyplýva platnosť nasledovného tvrdenia.

Veta 5.2.9. *Pre ľubovoľné dostatočne veľké n existuje formula s $2n$ premennými, v ktorej pre každé rozdelenie premenných na dve polovice I, existuje λn premenných v jednej polovici, ktoré su v klauzule s premennou z druhej polovice. Navyše graf izomorfný s touto formulou je bipartitný s partíciami veľkostí n, n a vrcholy v ľavej partícii majú stupeň $c < 20$.*

Teraz sme v stave, že máme už nejakú formulu A, ktorá spĺňa vetu 5.2.9, jedno pevné rozdelenie vstupu I a musíme ukázať, že vieme nájsť dostatočne veľkú *klamúcu množinu*. Pozrime sa teraz na formulu A ako na graf G, s ktorým je izomorfná. Viď obrázky 5.4 a 5.2, X a Y sú tak ako doteraz partície grafu G.

Veta 5.2.9 nám zaručuje, ze existuje taký podgraf G' grafu G, ktorý je bipartitný s partíciami S a T, partícia T je tvorená aspoň λn vrcholmi z $V \setminus I$ a každý vrchol z nej je spojený s aspoň jedným vrcholom z partície S, ktorá ma vrcholy z I. $G' = (V', E')$, $V' = S \cup T \subset V$, $E' = E \cap S \times T$, kde

$$S = \{u \mid \exists v \in V \setminus I \colon (u,v) \in E\} \cap (I \setminus Y)$$

$$T = \{v \mid \exists u \in I \colon (u,v) \in E\} \setminus (I \cup X)$$

Inými slovami S sú vrcholy z X, ktoré dostane prvý počítač, T sú vrcholy z Y, ktoré dostane druhý počítač, zároveň požadujeme, aby každý vrchol z T bol spojený s nejakým vrcholom z S.

Teraz vyslovíme pár viet o grafe G', ktoré použijeme neskôr na dôkaz existencie veľkej *klamúcej množiny*. Na to nám bude stačiť existencia dosť veľa klauzúl, ktoré sú akosi nezávislé. A každý počítač bude poznať hodnotu iba jednej premennej z týchto klauzúl. Potom budeme môcť ľubovoľne kombinovať samokomplementárne priradenia premenných na týchto hranách (klauzulách), tak ako sme to urobili pri dôkaze dolného odhadu pre konkrétnu funkciu, skôr v tejto kapitole.

Najprv graf G' ešte trochu upravíme. Môžeme predpokladať, že $|T| = \lambda n$, ak nie, môžeme ľubovoľné vrcholy odstrániť. Z toho následne vyplýva, $|S| \geq \lambda n/c$ a že ak $|S| > \lambda n$, môžeme nejaký vrchol z S odstrániť. Potom ale môžeme predpokladať $|S| \leq \lambda n$. Na základe týchto predpokladov môžeme urobiť nasledovnú úvahu: Počet vrcholov z Y, ktoré majú stupeň väčší ako $2c$, je najviac $\frac{\lambda nc}{2c} = \lambda n/2$. Lebo ináč by bol súčet stupňov vrcholov v jednej partícii väčší ako v druhej. Vyhoďme tieto vrcholy a zopakujme ešte raz predošlé úvahy. Dostaneme nasledovné vlastnosti grafu G':

- $|S| \geq \lambda n/2c$
- $|S| \leq \lambda n/2$
- $|T| = \lambda n/2$
- Stupeň každého vrcholu z T je najviac $2c$.

Ďalej nech $\lambda' = \lambda/2$

Veta 5.2.10. *V grafe G' existuje párenie o veľkosti $\lambda' n/c$*

Dôkaz. Podľa Königovej vety [Die05] nám stačí dokázať, že minimálne vrcholové pokrytie G' je aspoň $\lambda' n/c$. Ak v ľubovoľnom vrcholovom pokrytí vyberiem najviac k vrcholov z T, potom musíme ešte pokryť aspoň $\lambda' n - k$ hrán, ktoré vychádzajú zo zvyšných vrcholov z T, nejakými vrcholmi z S. Vrcholy z S majú ale stupeň najviac c, takže dostávame, že minimálne vrcholové pokrytie je väčšie ako $\min_k\{\frac{\lambda' n - k}{c} + k\} \geq \lambda' n/c$ □

Veta 5.2.11. *Nech $k \leq \lambda' n/3(c+1)c$, potom v G' existuje množina k hrán $M = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$ taká, že platí:*

$$\begin{array}{lcl} x_i & \in & S \\ y_i & \in & T \\ (x_i, y_j) & \in & E \text{ práve vtedy keď } i = j \\ \{x_i, y_i\} \cap \{x_j, y_j\} & = & \emptyset \text{ ak } i \neq j \end{array} \tag{5.10}$$

Dôkaz. Nech $m = \lfloor \lambda' n/c \rfloor$, $m' = \lfloor \lambda' n/3c^2 \rfloor$. Podľa vety 5.2.10, v G' existuje párenie $M' = \{(x'_1, y'_1), (x'_2, y'_2), \ldots, (x'_m, y'_m)\}$. Dôkaz vety urobíme matematickou indukciou. Pre $k = 0$ výber existuje. Nech $0 \leq k' < m'$, a nech existuje $M = \{(x''_1, y''_1), (x''_2, y''_2), \ldots, (x''_{k'}, y''_{k'})\}$.

Každá dvojica (x''_i, y''_i) nám odstráni maximálne $3c$ kandidátov pre výber doplňujúcej dvojice z párenia. Prvých c tvoria susedia vrchola x''_i, konkrétne hrany na ktorých tieto vrcholy ležia, druhých $2c$ tvoria susedia vrcholu y''_i. Musí teda ešte ostať aspoň
$m - 3ck' > \lfloor \lambda' n/c \rfloor - 3c\lfloor \lambda' n/3c^2 \rfloor =^5 \lambda' n/c - [\lambda' n/c] - 3c(\lambda' n/3c^2 - [\lambda' n/3c^2]) =$
$= [\lambda' n/c] + 3c[\lambda' n/3c^2] \geq 0$ kandidátov na doplnenie. Takže existuje nejaké y'_i, ktoré nie je v okolí žiadneho x''_j, a zároveň ani x'_i nie je v okolí žiadneho y''_j .
Potom $M = \{(x''_1, y''_1), (x''_2, y''_2), \ldots, (x''_{k'}, y''_{k'}), (x'_i, y'_i,)\}$, obsahuje o jeden prvok viac, čím sme dokázali indukčný krok. □

Tvrdenie 5.2.12. *Existuje klamúca množina $\mathcal{S}$ veľkosti $2^{\Omega(n)}$.*

Dôkaz. Nech $k = \lfloor \lambda' n/3c^2 \rfloor$, a $M = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$ je príslušná množina hrán, podľa vety 5.2.11. *Klamúcu množinu* $\mathcal{S}$ zostrojíme jednoducho, pre slovo $w \in \mathcal{S}$ platí, že *bity*, ktoré nereprezentujú žiadnu premennú x_i ani y_i, budú 1. Ostatné bity budú nastavené tak, aby *bit* pre x_i bol rôzny od y_i pre všetky i. Hodnota formuly A pre vstup $w \in \mathcal{S}$ je 1, lebo formula neobsahuje negáciu, a ak je nejaká premenná 0, tak je to buď x_i alebo y_i, ktoré sú spolu v klauzule, majú rôznu hodnotu a disjunkcia dvoch rôznych pravdivostných hodnôt je vždy pravdivá, navyše už nie sú v klauzule so žiadnou inou premennou x_j alebo y_j,kde $i \neq j$.

Z definície je dané, že x_i je v inej polovici vstupu ako y_i, teda ak skrížime nejaké dve rôzne slová z $\mathcal{S}$, dostaneme, že sa pre nejaké i hodnota $x_i = y_i$. Ak je $x_i = 0$, tak formula nie je splnená. Ak $x_i = 1$, potom v opačnom skríženie je $x_i = 0$. Slov v $\mathcal{S}$ môže byť $2^k = 2^{\Omega(n)}$. □

Dôsledok. *Použitím dôsledku 3.2.3 dostaneme $D^{\mathrm{BEST}}(A) \in \Omega(n)$.*

5.2.2 Obmedzené *2-CNF*

Táto časť obsahuje triviálny výsledok, a to, že ak v *2-CNF* formuli obmedzíme počet výskytov každej premennej na 2, tak dostaneme reprezentáciu veľmi ľahkých funkcií. Dôkaz je jednoduchý, totiž graf izomorfný s takouto formulou má komponenty kružnice alebo cesty. Vrcholy takéhoto grafu vieme rozdeliť tak, že najviac jeden komponent ostane rozdelený medzi dvoma polovicami vstupu. Keďže komponenty sú kružnice alebo cesty, toto rozdelenie

[5] $[x] = x - \lfloor x \rfloor$

môžeme urobiť tak, že najviac 2 klauzule nemajú oba vstupy v jednej polovici. Potom stačí poslať jednému účastníkovi 3 bity, hodnoty premenných, ktoré sú v klauzule s premennou, ktorú má druhý účastník a výsledok konjunkcie klauzúl, ktroých premenné má na vstupe tento účastník.

Kapitola 6

Záver

Podarilo sa nám dokázať tvrdenie, o ktorom sme predpokladali, že neplatí. Toto tvrdenie zaručuje existenciu lineárne veľkej *2-CNF* formule, ktorá reprezentuje funkciu s lineárnou komunikačnou zložitosťou. Na základe toho môžeme usúdiť, že z vysokej komunikačnej zložitosti nejakej funkcie nijako nemôže nevyplývať superlineárny dolný dohad pre velkosť logického obvodu, ktorým môžeme túto funkciu vypočítať.

Ďalšie tvrdenia, ktoré sme dokázali, prispievajú k poznatkom o vlastnostiach komunikačnej zložitosti a k fondu funkcií, ktorých komunikačnú zložitosť poznáme.

Literatúra

[AB79] Tarjan Robert E. Aspvall Bengt, Plass Michael F. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

[Die05] Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Heidelberg, third edition, 2005.

[EK97] Noam Nisan Eyal Kushilevitz. *Communication complexity*. Cambridge University Press, USA, 1997.

[Hro97] Juraj Hromkovič. *Communication complexity and parallel Computing*. Springer-Verlag, Berlin Heidelberg Germany, 1997.