Baigtinių Skaičiavimų Kombinatorika: Apatinių Įverčių Problema

Stasys Jukna

Habilitacinio darbo santrauka

(parašyta: 2006 m. balandžio 20 d.)

Disertacija ginta: Triero Universiteto Informatikos Fakultete
(Vokietija)
Disertacijos gynimo data: 1999 m. liepos 7 d.

Rezultatai

Pagrindiniai habilitacinio darbo rezultatai yra šie:

(1) Nelygybės $\mathbf{P} \neq \mathbf{NP} \cap \text{co–}\mathbf{NP}$ įrodymas skaičiavimo medžiams.
(2) Nelygybės $\mathbf{P} \neq \mathbf{NP} \cap \text{co–}\mathbf{NP}$ įrodymas read-once programoms.
(3) Eksponentiniai apatiniai įverčiai ribotoms binarinėms programoms: read-k programom, $(1, +k)$-programoms ir semantiniams tų programų variantams. Šitie rezultatai tiems skaičiavimo modeliams lieka iki šiol[1] geriausi.
(4) Naujas, informacijos teorija paremtas apatinių sudėtingumo įverčių gavimo metodas neribotom binarinėm programom.
(5) Monotoninių schemų sudėtingumo kriterijus. Tuo pačiu išspręsta A. Razborov'o 1986 metais Matematikų Kongrese (Berkeley) iškelta problema.
(6) Naujas, baigtinės ribos sąvoka paremtas apatinių sudėtingumo įverčių gavimo metodas riboto gylio schemoms.
(7) Eksponentinai apatiniai įverčiai schemoms su slenkstiniais elementais.
(8) Naujas, optimalių schemų nestabilumu paremtas apatinių įverčių metodas.
(9) Pirmi eksponentiniai įverčiai semantinės rezoliucijos įrodymų ilgiui.
(10) Pirmi aukšti (tiesiniai) apatiniai komunikacinio sudėtingumo įverčiai didesnio kaip logaritminio žaidėjų skaičiaus atveju.

[1] 2006 m. balandis

Publikacijos ir kiti fomalumai

Habilitacinis darbas parašytas algoritmų sudėtingumo teorijos srityje. Jis paremtas 1984-1999 metais autoriaus (po kandidatinės disertacijos gynimo 1980 metais) toje srityje[2] publikuotais 27 autoriaus moksliniais straipsniais. Iš jų

11 straipsnių žurnaluose, įtrauktuose į Mokslinės informacijos instituto (ISI) duomenų bazę:
- *Combinatorica* (Springer-Verlag) [31]
- *Computational Complexity* (Birkhäuser-Verlag) [5, 9, 32]
- *Discrete Applied Mathematics* (North-Holland) [33]
- *Doklady Akademii Nauk* (Nauka) [17]
- *Information Processing Letters* (North-Holland) [25, 30]
- *Theoretical Computer Science* (North-Holland), [16, 34]
- *Theoretical Informatics and Applications* (RAIRO) [23]

9 straipsnių referuojamuose periodiniuose leidiniuose:
- *Colloquia Mathematica Societatis János Bolyai* [10, 11]
- *DIMACS Series in Discrete Math. and Theoretical Comput. Sci.* (American Math. Society) [29]
- *Diskretnaja Matematika* (Nauka) [21]
- *Lecture Notes in Computer Science* (Springer-Verlag) [13, 15, 18, 19, 22]

Dalis habilitaciniame darbe pateiktų rezultatų yra įtraukti į monografiją

S. Jukna, *Extremal Combinatorics: With Applications in Computer Science*, Springer-Verlag, 2001, xvii + 375, ISBN 3-540-66313-4.

Monografijos rankraštis buvo įteiktas leidyklai 1998 m. vasario 12 dieną, bet pati monografija nebuvo teikiamos gynimui disertacijos dalis.

Tęsiant habilitaciniame darbe pradėtus tyrimus, po jo gynimo 2000-2006 metais paskelbti dar 6 straipsniai (ISI duomenų bazėje esančiuose) žurnaluose:

- *Combinatorics, Probability & Computing* (Cambridge University Press), 2 straipsniai
- *Information and Computation* (Springer-Verlag), 1 straipsnis
- *Information Processing Letters* (North-Holland), 2 straipsniai
- *SIAM Journal on Computing* (SIAM), 1 straipsnis

Habilitacijos komisiją sudarė 13 narių (habilituotų daktarų). Komisijos paskirti oponentai: Prof. Dr. D. Baum, Prof. Dr. Ch. Meinel, Prof. Dr. I. Wegener. Visos trys recenzijos buvo teigiamos. Disertacijos gynimo data: 1999 m. liepos 7 d. Slapto balsavimo rezultatas: 13:0.

[2]Publikacijos iki 1984 metų – kaip ir pati kandidatinė disertacija – buvo iš kitu sričių, neliečiamų habilitaciniame darbe: matematinės logikos taikymai ir loginių schemų patikimumo teorija.

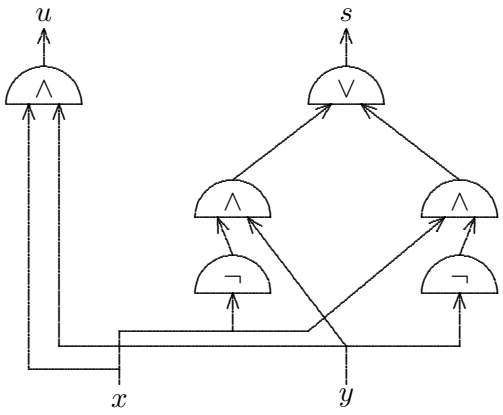
Turinys

Rezultatai 3
Publikacijos ir kiti fomalumai 5
1. Įvadas 7
2. Skaičiavimo medžiai 10
3. Read-once programos 12
4. Read-k programos 14
5. $(1, +s)$-programos 15
6. Entropinis metodas 16
7. Kontaktinės schemos be nuliniųkelių 17
8. Read-once kontaktinės schemos 19
9. Semantinės programos 20
10. Mažo gylio schemos 20
11. Schemos su slenkstiniais elementais 22
12. Monotoninės schemos 24
13. Optimalių schemų nestabilumas ir jo taikymai 25
14. Loginės įrodymos sistemos: rezoliuzija 26
15. Kommunikacinis sudėtingumas 28
Santraukoje cituota literatūra 32

1. Įvadas

Habiltaciniame darbe nagrinejama centrinė algoritmų sudėtingumo teorijos problema – taip vadinama *apatinių įverčių problema*. Jos esmė yra įrodyti, kad duota diskreti funkcija (paprastai, Bulio funkcija) reikalauja tam tikro kiekio elementarių operacijų. (Piešinyje 1 pateiktas vienos paprastos loginės schemos pavyzdys.) Įrodyti, kad duotą funkciją *galima* efektyviai realizuoti, yra dažniausiai paprasta: tam užtenka pateikti vieną konkretų algoritmą. Tačiau norint įrodyti, kad šis algoritmas yra optimalus, reikia įrodyti, kad *joks* šią funkciją realizuojantis algoritmas (jau egzistuojantis ar gal būt sukurtas kada nors ateityje) gali buti efektyvesnis.

Tarkim, kiek loginių operacijų (konjuncija $\wedge$, dizjunkcija $\vee$, neigimas $\neg$) reikia norint gauti dviejų skaičių $x, y \leq 2^n$ sandaugos $z = x \cdot y$ ar sumos $z = x + y$ dvejetainį kodą? Elementarus mokyklinis daugybos algoritmas rodo, kad sandaugai n^2 operacijų pakanka. Kita vertus, sumai užtenka $O(n)$, t.y. tiesinio skaičiaus operacijų. Ar iš tikrųjų sandaugai reikia daugiau operacijų nei sumai? Šis fundamentalus klausimas iki šiol lieka neatsakytas. Naudojant Fourier transfromaciją, pavyko įrodyti, kad sandaugai pakanka $O(n \log n \log\log n)$ operacijų. Tačiau niekam iki šiol nepavyko įrodyti, kad



PIEŠINYS 1. Loginė schema realizuojanti dvi Bulio funkcijas $s = x + y \pmod 2$ ir $u = x \wedge y$. Ši schema naudojama dvejetainiu skaičiu sumai realizuoti; jei i-tieji tu skaičiu bitai yra x ir y, tai s yra ju suma ir u yra kėlinio bitas.

bet kuris daugybos algoritmas privalo naudoti, tarkim, mažiausiai $10n$ operacijų!

Pagrindinis habilitaciniame darbe nagrinėjamos *apatinių* įverčių gavimo problemos sunkumas glūdi žodžiuose "bet kuris algoritmas." Tad apatinių įverčių problema – tai efektyvių algoritmų konkrečioms funkcijoms *neegzistavimo įrodymo* problema. Garsi "$\mathbf{P} = \mathbf{NP}$?" problema yra tik specialus šios (bendresnės) problemos atvejas. Grubiai tariant, klasė **NP** susideda iš visų funkcijų $f(x, y)$ su $x, y \leq 2^n$ ir reikšmėm 0 ("taip") ir 1 ("ne") tokių, kad gavus skaičius x ir y, yra lengva[3] patikrinti ar $f(x, y) = 1$. Pvz., lengva patikrinti ar duotas skaičius y dalina duotą skaičių x. Tuo tarpu klase **P** susideda iš visų funkcijų $f(x, y)$ tokių, kad gavus skaičių x, yra lengva patikrinti ar $f(x, y) = 1$ *bent vienam* skaičiui y. Aišku, kad pastarasis uždavinys nėra lengvesnis už pirmąjį, bet (bent iš prinzipo) gali būti sunkesnis. Tarkim, yra "aišku", kad padalinti du skaičius x, y neturėtų būti vienodai sunku, kaip nuspręsti ar duotas skaičius x yra pirminis. Lygybė $\mathbf{P} = \mathbf{NP}$ reikštų, kad abu šie uždaviniai yra vienodai sunkūs.

Norint įrodyti, kad $\mathbf{P} \neq \mathbf{NP}$, pakanktų surasti bent vieną funkciją $f \in \mathbf{NP}$, kurios negalima realizuoti polinominio dydžio logine schema; tada $f \notin \mathbf{P}$. Taigi, ši problema tikrai yra apatinių įverčių problemos atskiras atvejas. Be grynai matematinės šios problemos svarbos, ji yra svarbi ir praktikoje. Tarkim, žinoma kriptografinė sistema RSA yra pagrįsta prielaida, kad joks algoritmas yra pajėgus faktorizuoti duota skaičių polinominiame laike. Tačiau tai yra tik *prielaida* – kol nėra eksponentinio šios problemos

[3]"Lengva" čia reiškia polynominiame nuo kintamųjų skaičiaus n laike, ar naudojant polinominį skaičių loginių elementų.

skaičiavimo sudėtingumo apatinio įverčio matematinio *įrodymo*, ši (plačiai praktikoje paplitusi) kriptografinė sistema lieka nesaugi. Pavyzdžiui, ilgą laiką buvo manoma, kad joks efektyvus algoritmas negali nustatyti ar duotas skaičius yra pirminis, t.y. kad problema PRIMES reikalauja daugiau nei polinominio skaičiaus elementarių operacijų, t.y. kad PRIMES $\notin \mathbf{P}$. Bet visai nesenai buvo įrodyta,[4] kad tai netiesa!

Tokie pavyzdžiai rodo, kad ši – informatikoje gimusi, bet matematikoje užaugusi – algoritmų sudėtingumo apatinių įverčių problema iš tikrųjų yra gana sunki matematinė problema.[5] Ji buvo pradėta aktyviai tirti prieš maždaug 50 metų. Aišku, kad kiekvienai Bulio funkcijai $f : \{0,1\}^n \to \{0,1\}$ užtenka $O(2^n)$ elementų – tai trivialus *viršutinis* įvertis visoms funkcijoms. Vėliau Lupanov 1960 įrodė, kad užtenka net $O(2^n/n)$ elementų. Kita vertus, naudojant tikimybinį argumentą nesunku įrodyti, kad *beveik kiekvienos* Bulio funkcijos realizacijai reikia mažiausiai $\Omega(2^n/n)$ elementų. Kitaip sakant, Bulio funkcijų, realizuojamų polinominio dydžio schemom, dalis visų Bulio funkcijų klasėje yra nykstamai maža. Tai įrodė Shannon dar 1949 metais.

Tačiau iki šiol niekam nepavyko įrodyti, kad kuri nors *konkreti* (tarkim, priklausanti klasei **NP**) funkcija reikalauja daug elementų: nepaisant daugelio matematikų pastangų, aukščiausiu lieka apatinis įvertis[6] $3n-o(1)$ pasiektas Paul (1977), Schnorr (1980) ir Blum (1984).

Aukštesnius (kvadratinius) įverčius pavyko gauti tik *ribotom* schemom: kontaktinėm schemom (Nechiporuk 1966) ir Bulio formulėm (Khrapchenko 1971). Po šių rezultatų įvairios ribotų loginių schemų klasės pardėtos tirti ypač intensyviai. Šia kryptimi buvo gauta visa eilė svarių rezultatų: ribojant schemų strukturą pavyko įrodyti net *eksponentinius* apatinius įverčius tokiuose modeliuose, kaip kontaktinės schemos be nulinių kelių, "vienaskaitinės" (read-once) binarinės programos, monotoninės schemos, riboto gylio schemos ir kt. Žymiai svarbiau nei patys įverčiai yra jų gavimui sukurti nauji apatinių įverčių įrodymo *metodai*, tokie kaip:

(1) "bottlenecks counting" argumentas rezoliucijai ir monotoninėms schemoms,
(2) skaičiavimų "maišymo" metodas (the fusion method) monotoninėms schemoms ir algebrinėms programoms,
(3) atsitiktinių projekcijų metodas riboto gylio schemom,
(4) aproksimavimo metodas monotoninėm schemom,
(5) "skaldyk ir valdyk" principo variantai įvairiems binarinių programų modeliams,

[4]M. Agrawal, N. Kayal, N. Saxena, PRIMES is in **P**, *Annals of Mathematics* 160, No. 2, 781-793 (2004).

[5]Clay Mathematics Institute (Cambridge, Massachusetts) išrinko "$\mathbf{P} = \mathbf{NP}$?" problemą kaip vieną iš septynių svarbiausių praeito amžiaus matematinių problemų (žr. http://www.claymath.org/millennium/).

[6]2001 metais keturiems matematikams K. Iwama, O. Lachish, H. Morizumi ir R. Raz pavyko įrodyti šiek tiek didesnį įvertį $5n - o(1)$.

(6) entropinis metodas neribotom binarinėms programoms,
(7) "baigtinių ribų" metodas riboto gylio ir monotoninėm schemom.

Habilitaciniame darbe pateikiamas autoriaus 15 metų (tarp 1984 ir 1999) indėlis vystant pirmus keturius metodus (1)-(4); paskutiniai trys metodai (5)-(7) buvo pasiūlyti paties autoriaus. Indėlis pateikiamas gana konkrečiai – demonstruojama kaip siūlomi metodai dirba konkrečiuose algoritmų modeliuose. Todėl pats darbas yra suskaidytas į atskiras dalis, atitinkančias tuos metodus demonstruojančius algoritmų modelius:

(1) binariniai medžiai (decision trees);
(2) ribotos binarinės programos (branching programs);
(3) gylio-3 schemos (depth-3 ciruits);
(4) monotoninės schemos (monotone circuits);
(5) rezoliucija;
(6) komunikaciniai protokolai.

Toliau mes trumpai aprašysime tose dalyse pateikiamus rezultatus. Kadangi šis rašinys yra tik santrauka, mes apsiribosime tik pačiais tipiškiausiais rezultatais. Be to mes dažnai formuluosime rezultatus ne bendru pavidalu (kaip jie yra pateikiami atitinkamose publikacijose), o imsime tik jų atskirą, bet palygint lengvai suprantamą ir kitų sričių matematikams, atveją.

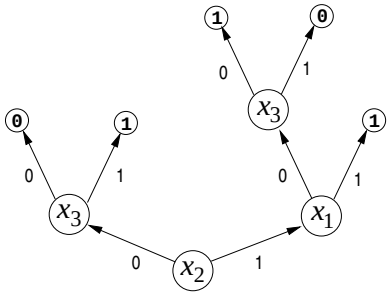
2. Skaičiavimo medžiai

Pirmose disertacijos dalyse nagrinejamas klasikinis Bulio funkcijų skaičiavimo modelis – binarinės programos (branching programs) ir jų specialus atvejas – skaičiavimo medžiai.

Skaičiavimo medis (decision tree) yra baigtinis orentuotas medis T, kurio kiekviena vidinė viršūnė turi du išeinančius lankus, pažymėtus 0 ir 1. Kiekvienai vidinei viršūnei (t.y. ne lapui) priskirtas kuris nors vienas kintamasis x_i, $1 \leq i \leq n$. Lapams priskirtos reikšmės 0 ar 1. Viena vidinė viršūnė paskelbiama *pradine*.

Gavus dvejetainį vektorių $a = (a_1, \ldots, a_n) \in \{0,1\}^n$ skaičiavimas prasideda pradinėje viršūnėje ir vyksta šitaip: jei skaičiavimas pasiekia viršūnę pažymetą kintamuoju x_i, tai toliau skaičiavimo kelias eina lanku pažymėtu $a_i \in \{0,1\}$. Kadangi medis yra baigtinis, tokiu būdu mes paieksime kurį nors lapą – to lapo žymė (bitas 0 ar 1) ir yra skaičiavimo medžio reikšmė $T(a)$. Tuo būdu medis skaičiuoja atitinkamą Bulio funkcija $T : \{0,1\}^n \to \{0,1\}$. Medžio *dydis* – tai viršūnių skaičius joje. Jei medyje yra nepažymėtų viršūnių, tada pasiekus tokią viršūnę skaičiavimo kelią galima pratęsti bet kuriuo iš išeinančių lankų. Tokiu atveju skaičiavimo medis yra *nedeterminuotas* (non-deterministic) ir $T(a) = 1$ tada ir tik tada, kai bent vienas skaičiavimas veda į lapą su žyme 1

Tegu $\mathrm{dt}(f)$ žymi mažiausią Bulio funkciją f realizuojančio (determinuoto) medžio viršūnių skaičių.



PIEŠINYS 2. Skaičiavimo medis; įvedus vektorių $(0, 1, 0)$ jis duoda atsakymą 1.

Darbe (Jukna *et al.* 1997) mes nagrinėjame tokį problemos "$\mathbf{P} = \mathbf{NP} \cap$ co–$\mathbf{NP}$?" variantą skaičiavimo medžiams:

Problema 1. Tarkim, Bulio funkcija f *ir* jos neigimą $\neg f$ galima realizuoti *nedeterministiniais* skaičiavimo medžiais su N viršūnių. Ar tada galioja $\mathrm{dt}(f) \leq N^c$ kuriai nors konstantai $c > 0$?

Ši problema buvo atvira daugelį metų. Pakartotinai tos problemos svarbą taip vadinamoje "besimokančių algoritmų" teorijoje (algorithmic learning theory) pabrėžė Ehrenfeucht and Haussler (1989). Tame darbe jie įrodė *viršutinį* įvertį $\mathrm{dt}(f) \leq n^{O(\log^2 N)}$.

Naudojant algebrinį (spektralinį) argumentą darbe Jukna *et al.* (1997) mes įrodom *apatinį* įvertį $\mathrm{dt}(f) \geq 2^{\Omega(\log^2 N)}$. Tuo pačiu mes įrodėme, kad skaičiavimo medžių modelyje sudėtingumo klases $\mathbf{P}$ ir $\mathbf{NP} \cap$ co–$\mathbf{NP}$ skiriasi, t.y. čia galioja $\mathbf{P} \neq \mathbf{NP} \cap$ co–$\mathbf{NP}$ ir tuo pačiu $\mathbf{P} \neq \mathbf{NP}$.

Įrodymas naudoja funkcijų Fourier transformaciją. Tam tikslui mes vietoj kubo $\{0,1\}^n$ imame kubą $\{-1,+1\}^n$ ir vietoj funkcijų $f : \{0,1\}^n \to \{0,1\}$ nagrinejame funkcijas $f : \{-1,+1\}^n \to \{-1,+1\}$. Tokių funkcijų aibė duoda mums dimensijos 2^n tiesinę erdvę su skalarine vektorių daugyba $\langle f, g\rangle = 2^{-n} \sum_x f(x)g(x)$. Monomai $X_S = \prod_{i \in S} x_i$ duoda mums to erdvės ortonoralią bazę. Taigi, kiekvieną funkcija $f : \{-1,+1\}^n \to \{-1,+1\}$ galima vienareiksmiškai išreikšti kaip sumą $f = \sum_S \hat{f}(S) X_S$; čia

$$\hat{f}(S) = \langle f, X_S\rangle = 2^{-n} \sum_{x \in \{-1,+1\}^n} f(x) X_S(x)$$

yra taip vadinamas S-tasis funkcjos f Fourier koeficientas. Tą vienareikšmišką funkcijos f išraišką kaip monomų suma galima gauti iš bet kokio funkciją f išreiškiančio realaus polinomo naudojant tapatybes $x_i^2 = 1$.

Musų pagrindinis techninis rezultatas yra teorema, duodanti funkciją f realizuojančio skaičiavimo medžio dydžio $\mathrm{dt}(f)$ apatinį įvertį tos funkcijos Fourier koeficientų terminais.

Teorema 2 (Jukna *et al.* 1997)**.** *Kiekvienai Bulio funkcijai* $f(x_1, \ldots, x_n)$ *ir kiekvienam poaibiui* $S \subseteq \{1, \ldots, n\}$ *galioja:*

$$\mathrm{dt}(f) \geq 2^{|S|} \cdot \sum_{T \supseteq S} |\hat{f}(T)|.$$

Naudojant šią teoremą mums pavyko įrodyti, kad deterministiniai skaičiavimo medžiai kai kurioms (konkrečioms) Bulio funkcijos f reikalauja $N^{\Omega(\log N)}$ viršūnių, tuo tarpu kai f ir jos neigimą $\neg f$ galima realizuoti nedeterminuotais dyžio N skaičiavimo medžiais. Tuo pačiu Problema 1 buvo išspręsta. Be to, pagal aukščiau minėtą Ehrenfeucht and Haussler (1989) rezultatą, šis mūsų nustatytas skirtumas yra beveik maksimalus.

Įdomu pažymėti, kad šis mūsų rezultatas kontrastuoja su kitu žinomu rezultatu, liečiančiu problemą "$\mathbf{P} = \mathbf{NP} \cap$ co–$\mathbf{NP}$?" skaičiavimo medžiams. Mes matuojame tokių medžių sudėtingumą juose esančių viršūniu skaičiumi. Kitas natūralus medžių matas yra ju *gylis*, t.y. lankų skaičius ilgiausiame kelyje iš pradinės viršūnes iki kurio nors lapo. Net trys matematikų grupes nepriklausomai ir beveik vienu metu (apie 1990 metus) įrodė, kad tokiu atveju galioja lygybė $\mathbf{P} = \mathbf{NP} \cap$ co–$\mathbf{NP}$. Tad buvo tikimasi, kad ši lygybė turėtų galioti ir imant viršūnių skaičių kaip sudėtingumo matą. Todėl mūsų rezultatas buvo gana netikėtas.

3. Read-once programos

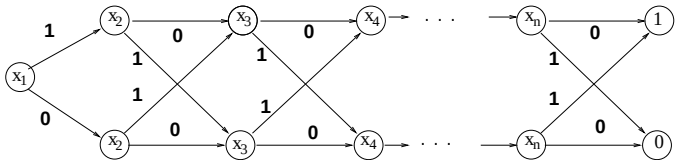
Binarinė programa (branching program) – toliau paprasčiausiai *programa* – yra skaičiavimo medžio apibendrinimas. Vienintelis skirtumas yra tas, kad dabar vietoj medžio leidžiama imti bet kokį orentuotą grafą be ciklų.

Pastebėkime, kad skaičiavimo medyje nėra jokios prasmės kuriame nors kelyje iš pradinės viršūnės iki lapo vieną ir tą patį kintamąjį x_i testuoti daugiau nei vieną karta. Kitaip yra binarinėse programose: čia leidžiant tą patį kintamąjį testuoti keletą kartų galima žymiai (net eksponentiškai) sumažinti bendrą programos viršūnių skaičių. Kadangi bendroms (neribotoms) programoms aukščiausiu lieka dar 1966 metais Nechiporuk gautas įvertis $\Omega(n^2/\log^2 n)$, buvo pradėta tirti tų kintamųjų pakartotinių testavimų įtaką programos dydžiui.

Binarinė programa $P(x_1, \ldots, x_n)$ yra *read-once* programa, jei bet kuriame kelyje iš pradinės viršūnės iki lapo kiekvienas kintamasis x_i sutinkamas daugiausiai vieną kartą. Pastebėkime, kad tokios programos – tai skaičiavimo medžiai, kurių izomorfiški pomedžiai yra "sulipdyti". Tad read-once programos yra pirmas natūralus skaičiavimo medžių apibendrinimas.

Kita vertus, read-once programos gali būti žymiai ekonomiškesnės už skaičiavimo medžius. Imkim pavyzdžiui tokią Bulio funkciją:

$$f(x_1, \ldots, x_n) = x_1 + x_2 + \cdots + x_n \pmod 2.$$



PIEŠINYS 3. Read-once programa funkcijai $x_1 + x_2 + \cdots + x_n \pmod 2$

Triviali šios funkcijos read-once programa turi tik $2n + 1$ viršūnių (žr. Piešinį 3), tuo tarpu kai bet kuris binarinis medis šiai funkcijai privalo turėti net $2^{\Omega(n)}$ viršūnių (kadangi pakeitus bet kurį vieną vektoriaus koordinatę keičiasi ir funkcijos reikšmė). Tad nenuostabu, kad didelių (eksponentinių) apatinių įverčių read-once programoms įrodymas yra sunkesnis uždavinys nei binariniams medžiams.

Dabe (Jukna 1986) be kitų rezultatų buvo įrodytas toks bendras apatinis įvertis read-once programom. Bulio funkcija $f(x_1, \ldots, x_n)$ yra vadinama *m-sumaišyta* (m-mixed), jei paėmus bet kuriuos m kintamųjų ir įstačius jų vietoj konstantas 0 ar 1 bet kuriais dviem skirtingais būdais gauname dvi skirtingas Bulio funkcijas (nuo likusių nefiksuotų kintamųjų).

Teorema 3 (Jukna 1986)**.** *Jei Bulio funkcija f yra m-sumaišyta, tai kiekviena ją skaičiuojanti read-once programa privalo tureti mažiausiai $2^m - 1$ viršūnių.*

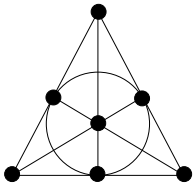
Daugelis iki šiol žinomų (ir kitų autorių įrodytų) apatinių įverčių konkrečioms Bulio funkcijom buvo gauti naudojant šį kriterijų. Vėliau Simon ir Szegedy (1993) apibendrino šį rezultatą.

Darbe (Jukna *et al.* 1997) mes nagrinėjome Problemos 1 analogą read-once programoms ir įrodėme, kad čia vėlgi galioja nelygybė $\mathbf{P} \neq \mathbf{NP} \cap \text{co–}\mathbf{NP}$. Šio rezultato įrodymui pagrindinė problema buvo sukonstruoti tinkamą Bulio funkciją. Mes ją sukonstravom naudodami projektyvines plokštumas (projective planes).

Kaip žinoma, projektyvinėje plokštumoje $PG(2, q)$, kur q yra pirminis skaičius ar tokio skaičiaus laipsnis, mes turime $n = q^2 + q + 1$ "taškų" $V = \{1, \ldots, n\}$ ir n tų taškų poaibių $L_1, \ldots, L_n$; šie poaibiai vadinami "tiesiemis". Kiekviena tiesė susideda iš lygiai $q + 1$ taškų, bet kurios dvi tieses kertasi lygiai viename taške ir per kiekviena tašką eina lygiai $q + 1$ tiesių (žr. piešinį 4).

Aibė taškų $S \subseteq V$ vadinama *blokuojančia aibe*, jei ji kerta kiekvieną tiesę. Taigi, tiesės yra mažiausios blokuojančios aibės. Nagrinėkime Bulio funkciją $f : \{0,1\}^n \to \{0,1\}$ tokią, kad $f(a_1, \ldots, a_n) = 1$ tada ir tik tada kai aibė vektoriaus a nenulinių koordinačių $S = \{i \in V : a_i = 1\}$ turi ne daugiau kaip $q + \sqrt{q}$ elementų ir yra blokuojanti.

Teorema 4 (Jukna *et al.* 1997)**.** *Funkciją f ir jos neigimą $\neg f$ galima realizuoti nedeterminuotomis dydžio $O(n^{5/2})$ read-once programomis, tačiau bet*



PIEŠINYS 4. Fano plokštuma $PG(2,2)$ su septyniom tiesėm ir trim taškais kiekvienoje tiesėje.

kuri determinuota read-once programa šiai funkcijai privalo turėti mažiausiai $2^{\Omega(\sqrt{n})}$ *viršūnių.*

Apatinis įvertis čia vėlgi gautas naudojant Teoremą 3.

Be kita ko, Teorema 4 rodo, kad ir read-once programų atveju nedeterministinės programos gali būti net eksponentiškai ekonomiškesnės negu deterministinės. Nepaisant to, darbe (Jukna 1986) buvo įrodytas bendras apatinis įvertis ir *nedetereminuotom* read-once programoms. Jis naudoja tokį Bulio funkcijų f matą. Tegu $T_r(f)$ yra mažiausias skaičius t, kuriam egzistuoja ilgio r monomai $K_1, \ldots, K_t$ tokie, kad $f \leq K_1 \vee \ldots \vee K_t$. Taip pat, tegu $d(f)$ žymi minimalų Hamming atstumą tarp bet kurių dviejų vektorių aibėje $f^{-1}(1)$, t.y. $d(f)$ yra minimalus skaičius d toks, kad bet kurie du vektoriai $a \neq b \in f^{-1}(1)$ skiriasi maziausiai d koordinatėse.

Teorema 5 (Jukna 1986)**.** *Kiekviena Bulio funkciją* f *realizuojanti nedeterministinė read-once programa privalo turėti mažiausiai* $T_{d(f)-1}(f)$ *viršūnių.*

Šios teoremos pagalba darbuose (Jukna 1986,1988a,1988b) buvo gauti pirmi eksponentiniai apatiniai įverčiai nedeterminuotoms read-once programoms. Veliau darbe (Jukna–Razborov 1998) ši teorema buvo išplėsta taip vadinamom "semantinėm" programom. Be to, naudojant tikimybinį argumentą buvo įrodyta, kad ši teorema duota eksponentinius apatinius įverčius visai eilei Bulio funkcijų $f(x_1, \ldots, x_n)$: tam pakanka, kad $m(f) \gg n/d(f)$, kur parametras $m(f)$ apibrėžiamas kaip mažiausias skaičius funkcijos f kintamųjų, kuriuos užfiksavus funkcija virsta trivialia funkcija lygia konstantai 0.

Lema 6 (Jukna–Razborov 1998)**.** *Tegu* $d(f) = d$ *ir* $m(f) = m$*. Tada*

$$T_d(f) = 2^{\Omega(md/n)}.$$

4. READ-k PROGRAMOS

Binarinė programa $P(x_1, \ldots, x_n)$ yra *read-k* programa, jei bet kuriame kelyje iš pradinės viršūnės iki lapo kiekvienas kintamasis x_i sutinkamas daugiausiai k kartų. Tokios programos yra natūralus read-once programų apibendrinimas (kur $k = 1$).

Darbe (Jukna 1995) pateikiamas bendras apatinių įverčių gavimo metodas nedeterminuotoms read-k programoms. Mes čia šio kriterijaus nekartosime. Vietoj to pateiksime viena konkretų jo taikymą rodantį, kad tam tikrų tiesinių "save taisančių" (self-correcting) kodų charakteristinių Bulio funkcijų negalima realizuoti polinominio dydžio read-k programomis.

Su kiekviena $m \times n$ matrica $A = \{a_{ij}\}$, kur $a_{i,j} \in \{0,1\}$, galima susieti Bulio funkciją

$$f_A(x_1,\ldots,x_n) = \bigwedge_{i=1}^{m} \left(1 \oplus \left(\bigoplus_{j=1}^{n} a_{ij}x_j\right)\right).$$

Tai yra, $f_A(x_1,\ldots,x_n) = 1$ tada ir tik tada kai vektorius $(x_1,\ldots,x_n)$ yra ortogonalus (kūne GF(2)) visoms matricos A eilutėms.

Imkim dabar gerai žinomą tiesinį BCH-kodą $C \subseteq GF(2)^n$, kuriame bet kurie du vektoriai skiriasi maziausiai $d = \Theta(\sqrt{n}/k^k)$ bitų. Tegu A yra to kodo "parity-check" matrica. Tada f_A yra kodo C charakteristinė funkcija: $x \in C \iff f_A(x) = 1$. Šiai funkcijai galioja šitokia

Teorema 7 (Jukna 1995)**.** *Jei $k = o(\log n/\log\log n)$, tai kiekviena nedeterminuota read-k programa funkcijai f_A privalo turėti $2^{\Omega(\sqrt{n})}$ viršūnių.*

Iki šiol[7] niekam nepavyko gauti eksponentinion apatinio įverčio tokiom programom kai $k \geq \log n$.

5. $(1,+s)$-PROGRAMOS

Read-k programose yra reikalaujama, kad kiekviename kelyje iš pradinės viršūnės iki lapo kiekvienas kintamasis x_i gali būti testuojamas daugiausiai k kartų. Dabar nagrinėkime kiek kitokį apribojimą: reikalaukime, kad kiekviename skaičiavimo kelyje[8] iki s kintamųjų gali būti testuojami kiek norima kartų; kiekvieną iš likusių $n-s$ kintamųjų galima testuoti daugiausiai vieną kartą. Tokios programos literatūroje vadinamos $(1,+s)$-*programomis*.

Pastebėkima, kad bet kuri programa su n kintamųjų yra $(1,+s)$-programa su $s \leq n$. Taigi, apatiniai įverčiai $(1,+s)$-programoms yra tuo stipresni kuo s yra didesnis.

Darbe (Jukna–Razborov 1998) buvo įrodytas toks bendras apatinių įverčių $(1,+s)$-programoms kriterijus.

---

[7] 2006 m. balandis.

[8] Ne kiekvienas kelias programos grafe atitinka skaičiavimo kelią: tokie yra, pavyzdžiui, keliai, kuriuose sutinkami lankai atitinkantys testus $x_i = 0$ ir $x_i = 1$; tokie keliai nėra realizuojami, nes tas pats bitas x_i negali tuo pačiu metu būti lygus 0 ir 1. Tačiau, kaip parodyta darbe (Jukna 1989), tokių "nereikalingų" kelių buvimas gali net eksponentiškai padidinti programos galią. Pastebėkime, kad $(1,+s)$-programose jokių apribojimų tokiems "nereikalingiems" keliams nėra.

Teorema 8 (Jukna–Razborov 1998)**.** *Tegu $d(f) = d$ ir $m(f) = m$. Tada bet kuri Bulio funkciją f realizuojanti $(1, +s)$-programa privalo turėti*

$$2^{(\min\{d,\ m/(s+1)\}-1)/2}$$

viršūnių.

Naudojant šį kriterijų tame pačiame darbe buvo įrodyta, kad kai kurių Bulio funkcijų $f(x_1, \ldots, x_n)$, būtent – charakteristinių tam tikrų tiesinių kodų funkcijų – negalima realizuoti polinominio dydžio $(1, +s)$-programomis, jeigu[9] $s = o(n/\log n)$.

6. Entropinis metodas

Darbe (Jukna–Žák 1998) mes pasiūlėme tokį apatinių sudėtingumo įverčių *neribotom* binarinėm programom gavimo metodą. Šis metodas paremtas informacijos tekėjimo programoje analize. Pagrindinė idėja yra matuoti programos "netikrumo kiekį" apie funkcijos reikšmę $f(a)$ tos reokšmės skaičiavimo metu. Gavusi vektorių $a = (a_1, \ldots, a_n) \in \{0,1\}^n$, programa pradeda skaičiavimą pradinėje viršūnėje. Tuo metu ji dar nieko apie vektorių a nežino. Kiekviename tolesniame žingsnyje programa tikrina tam tikrų koordinačių a_i reikšmes. Paklaususi ar $a_i = 0$ (ir gavusi atsakymą) programa gauna vieną papildomą informacijos bitą apie visą vektorių a. Tačiau ši informacija gali vėliau būti prarasta kurioje nors viršūnėje v, jei egzistuoja kitas vektorius b toks, kad $b_i \neq a_i$, skaičiavimo kelias $comp(b)$ pasiekia tą pačią viršūnę v ir arba i-toji koordinatė x_i vėl testuojama viršūnėje v arba po šios viršūnės abu skaičiavimai seka vieną ir tą patį kelią iki kurio nors lapo. Abiem atvejais programa yra vėl "netikra" apie koordinatę a_i viršūnėje v: pirmu atveju ji testuoja i-tąją koordinatę dar kartą, o antruoju jos ansktesnė informacija apie tos koordinatės reikšmę tampa nebereikalinga.

Tam, kad matematiškai išreikšti tą programos "netikrumo" kiekį, darbe (Jukna–Žák 1998) mes įvedėme skaičiavimų "entropijos" sąvoką. Jos idėja yra tokia.

Tegu $P = (V, E)$ yra kokia nors programa, skaičiuojanti duotą Bulio funkciją $f : \{0,1\}^n \to \{0,1\}$. Naudojant funkcijos sąvybes parenkame atitinkamą vektorių iš $\{0,1\}^n$ paskirstymą (distribution) $\varphi : \{0,1\}^n \to V$ tarp programos viršūnių. Tai yra pagrindinis "kūrybinis aktas" – jis priklauso nuo to, kokią Bulio funkciją mes nagrinėjame. Tokį paskirstymą φ galima apibrėžti davus taisyklę, pagal kurią skaičiavimai $comp(a)$ turi būti sustabdomi tam tikrose programos viršūnėse.

Kiekvienas paskirstymas $\varphi : \{0,1\}^n \to V$ duoda mums viso kubo $\{0,1\}^n$ suskaidymą į nesikertančius blokus $F_v = \varphi^{-1}(v)$, $v \in V$; kai kurie iš tų blokų gali žinoma būti tušti. Vieno bloko F_v entropiją apibzėžiame kaip vidutinį tos vektorių aibės *išskaidančio medžio* gylį; toks medis yra paprasčiausiai

[9] Visai nesenai man pavyko pagerinti šį rezultatą iki $s = \Omega(n)$: S. Jukna, Expanders and time restricted branching programs, *Thereoretical Computer Science* (įteikta 2005).

skaičiavimo medis toks, kad kiekvieną jo lapą gali pasiekti tik vienas aibės F_v vektorius. Galų gale mes apibrėžiame paskirstymo φ *entorpiją* $H(\varphi)$ kaip vidutinę visų blokų $F_v = \varphi^{-1}(v)$ su $v \in V$ entropiją. Tada gllioja tokia

Teorema 9 (Jukna–Žák 1998)**.** *Tegu $P = (V, E)$ yra bet kokia binarinė programa. Tada kiekvienam paskirstymui $\varphi : \{0,1\}^n \to V$ galioja*

$$|V| \geq 2^{n-H(\varphi)}.$$

Pagal šią teoremą, tam kad gauti gerą apatinį viršūnių skaičiaus $|V|$ įverti, pakanką gauti gera viršutinį entropijos $H(\varphi)$ įvertį. Pačios teoremos įrodyme mes naudojame tokią, gerai Informacijos Teorijoje žinomą Kraft-McMillan nelygybę: jei $p_1, \ldots, p_k$ yra kurio nors binarinio medžio kelių nuo pradinės viršūnės iki jo lapų ilgiai, tai galioja $\sum_{i=1}^{k} 2^{-p_i} \leq 1$.

Darbe (Jukna–Žák 1998) mes demonstruojame šį metodą taip vadinamų "balancuotų" programų klasėje. Tos programų klasės mes čia formaliai neapibrėžinėšime, tik pažymėsime, kad šios programos yra galingesnės už *visus* anksteniuose skyriuose nagrietus modelius. Pavyzdžiui, mes jau minėjome, kad tam tikrų tiesinių kodų charakteristinių funkcijų negalima realizuoti polinominio dyžio read-k programomis bei $(1, +s)$-programomis. Kita vertus, visas tokias funkcijas galima realizuoti balancuotomis programins naudojant tik kvadratinį viršūnių skaičių.

Tad aišku, kad balancuotoms programoms iki šiol žinomi metodai negali duoti aukštų apatinių įverčių. Naudojant Teoremą 9 mums visgi pavyko tokius įverčius gauti. Tam tikslui mes imame gerai žinomą Bulio funkciją $\textsc{Clique}_n(X)$. Ši funkcija yra svarbi problemos "$\mathbf{P} = \mathbf{NP}$?" kontekste: jei $\textsc{Clique}_n \in \mathbf{P}$ tai $\mathbf{P} = \mathbf{NP}$. Ši funkcija atpažįsta ar duotas grafas G su n viršūnių turi tam tikro dydžio kliką (pilną pografį). Tai yra, aibė X susideda iš $\binom{n}{2}$ kintamųjų $x_{i,j}$ tokių, kad $x_{i,j} = 1 \iff$ duotame grafe G yra lankas, jugiantis viršūnes i ir j. Funkcija $\textsc{Clique}_n$ akzeptuoja dutą grafą tada ir tik tada, kai šis grafas turi pilną pografį su $\sqrt{n}$ viršūnių.

Teorema 10 (Jukna–Žák 1998)**.** *Bet kuri balancuota programa funkcijai $\textsc{Clique}_n$ privalo turėti $2^{\Omega(\sqrt{n})}$ viršūnių.*

7. Kontaktinės schemos be nulinių kelių

Darbuose (Jukna 1986, 1988a, 1988b, 1989) nagrinėjamas kontaktinių schemų modelis. Istoriškai šis modelis buvo pradėtas tirti žymiai anksčiau nei binarinės programos (jau 1949 metais Shannon gavo pirmus rezultatus tokioms schemoms). Šis modelis yra bendresnis už nedeterminuotas binarines programas ir yra šiuolaikinių elektroninių schemų prototipas.

Kontaktinė schema yra neorentuotas grafas $G = (V, E)$ kurio kiekvienam lankui $e \in E$ priskirtas kuris nors kintamasis x_i ar jo neigimas $\neg x_i$. Kurios nors dvi viršūnės s ir t yra paskelbiamos pradine ir galine (source and target). Kiekvienas dvejetainis vektorius $a = (a_1, \ldots, a_n) \in \{0,1\}^n$ duoda mums

grafo G pografį $G_a = (V, E_a)$ su $E_a \subseteq E$, turintį tik tuos grafo G lankus, kurių žymės yra suderintos su vektoriaus a reikšmėmis. Tai yra, jei lanko žymė yra x_i (atitinkamai, $\neg x_i$), tai šis lankas lieka pografyje G_a tada ir tik tada, kai $a_i = 1$ (atitinkamai, kai $a_i = 0$). Tokia schema skaičiuoja Bulio funkciją $f_G : \{0,1\}^n \to \{0,1\}$ natūraliu būdu: $f_G(a) = 1 \iff$ pografyje G_a yra bent vienas kelias iš pradinės viršūnės s į galinę viršūnę t.

Tokia schema vadinama "kontaktine", kadangi ji dirba kaip įprastos kontaktinės elektroninės schemos. Lankų žymes galima interpretuoti kaip "kontaktus". Gavus vektorių a, kontaktas $y_i \in \{x_i, \neg x_i\}$ yra "pralaidus", jei $y_i(a) = 1$, t.y. jei $y_i = x_i$ ir $a_i = 1$ arba $y_i = \neg x_i$ ir $a_i = 0$. Viršūnes s ir t galima interpretuoti kaip teigiamą polių ir neigiamą polių. Tada $f_G(a) = 1$, jei tarp polių srovė teka, ir $f_G(a) = 0$, jei srovė neteka.

Nulinis kelias schemoje G yra s-t kelias, kuriame sutinkamas koks nors kintamasis x_i ir jo neigimas $\neg x_i$. Aišku, kad tokie keliai yra nepralaidūs – jie nepraleidžia nei vieno vektoriaus a. Tad iš pirmo žvilgsnio tokie keliai turėtų būti nereikalingi – funkcijos $f_G(x)$ reikšmei jie neturi jokios įtakos. Tačiau – kaip įrodyta darbe (Jukna 1989) – tokie keliai gali būti labai naudingi norint sumažinti schemos dydį, t.y. bendrą kontaktų (lankų) skaičių. Tą faktą mes nustatėme naudodami tokį bendrą apatinį įvertį schemoms be nulinių kelių.

Aibės $A \subseteq \{0,1\}^n$ *k-tasis laipsnis* $\deg_k(A)$ yra maksimalus skaičius jos vektorių, visi iš kurių turi vienetus kuriose nors k koordinatėse. Tai yra, $\deg_k(A)$ yra maksimalus skaičius t, kuriam egzistuoja $B \subseteq A$, $|B| = t$ ir $S \subseteq \{1, \ldots, n\}, |S| = k$ tokie, kad $a_i = 1$ visiems $a \in B$ ir visiems $i \in S$. Vektoriaus $a \in \{0,1\}^n$ *svoris* yra jo nenulinių koordinačių skaičius $|a|$, t.y. $|a| = |\{i : a_i = 1\}|$ Aibė $A \subseteq \{0,1\}^n$ yra *k-homogeninė*, jei $|a| = k$ visiems $a \in A$.

Teorema 11 (Jukna 1989)**.** *Tegu $A \subseteq \{0,1\}^n$ yra k-homogeninė aibė. Tada kiekviena kontaktinė schema be nulinių kelių skaičiuojanti aibės A charakteristinę funkciją privalo turėti mažiausiai*

$$\max_{s:s\le k} \frac{|A|}{\deg_s(A) \cdot \deg_{k-s}(A)}$$

kontaktų.

Darbe (Jukna 1989) ši teorema iliustruojama tokiai konkrečiai Bulio funkcijai $EPM_n(X)$, kuri vėliau tapo žinoma kaip "exact perfect matching" funkcija. Jos kintamieji sudaro $m \times m$ matricą $X = \{x_{ij}\}$ su $m = \sqrt{n}$. Tada $EPM_n(X) = 1 \iff$ kiekvienoje matricos X eilutėje ir kiekviename jos stulpelyje yra lygiai vienas nenulinis elementas. Aibė $A = EPM_n^{-1}(1)$ yra m-homogeninė ir turi lygiai $n!$ elementų. Be to nesunku matyti, kad $\deg_s(A) = (m-s)!$ visiems $0 \le s \le m$. Imant $s = m/2$, Theorema 11 duoda mums tokį apatinį įvertį.

Teorema 12 (Jukna 1989)**.** *Kiekviena kontaktinė schema be nulinių kelių skaičiuojanti EPM_n privalo turėti mažiausiai $2^{\Omega(\sqrt{n})}$ kontaktų.*

8. Read-once kontaktinės schemos

Kontaktinė schema vadinama *read-once*, jeigu kiekviename pralaidžiame jos kelyje (t.y. kelyje neturininčiame kintamojo kartu su jo neigimu) kiekvienas kintamasis sutinkamas daugiausiai vieną kartą. Taigi, read-once kontaktinės schemos yra nedeterminuotų read-once programu praplėtimas. Kad šitas praplėtimas yra iš tikrųjų esminis buvo parodyta darbe (Jukna 1992) tokiu rezultatu.

Mes jau žinome, kad praeitame skyriuje apibrėžta Bulio funkcija EPM_n (exact perfect matching) reikalauja eksponentinio dydžio nedeterminuotų read-once programu bei kontaktinių schemų be nulinių kelių. Kita vertus galia tokia

Teorema 13 (Jukna 1992)**.** *Funkciją EPM_n galima realizuoti read-once kontaktine schema su $O(n^3)$ kontaktų.*

Šis rezultatas nustebino daugelį specialistų, nes iki tol buvo manoma, kad kontaktinės schemos negali būti žymiai galingesnės už binarines programas.

Kita vertus, pati teoremos įrodymo idėja yra gana paprasta. Būtent, mes konstruojame reikalaujamą read-once kontaktinę schemą funkcijai EPM_n pagal formulę $P = P_1 \wedge P_2$. kur

$$\begin{aligned} P_1(X) &= \bigwedge_{i=1}^{m} \bigvee_{j=1}^{m} x_{ij}, \\ P_2(X) &= \bigwedge_{j=1}^{m} \bigvee_{k=1}^{m} \bigwedge_{\substack{i=1 \\ i\neq k}}^{n} \neg x_{ij} \end{aligned}$$

Pastebėkime, kad $P_1(X) = 1 \iff$ kiekviena matricos X eilutė turi bent vieną vienetą, ir $P_2(X) = 1 \iff$ kiekvienas matricos X stulpelis turi bent $n-1$ nulį. Taigi, $P(X) = 1 \iff EPM_n(X) = 1$.

Teorema 13 rodo, kad read-once kontaktinės schemos turi nelauktai didelę galią. Ir iš tikro niekam iki šiol[10] dar nepavyko įrodyti eksponentinio įverčio tokioms schemoms. Taigi, šitas skaičiavimo modelis yra pats "silpniausias" nedeterministinis modelis, kuriam nepavyksta gauti aukštų apatinių įverčių.

[10] 2006 m. balandis

9. Semantinės programos

Tam, kad geriau suprasti ką tik minėtų read-once kontaktinių schemų (bei kitų aukščiau minėtų ribotų programų) funkcionavimą, darbe (Jukna–Razborov 1998) mes įvedėme naują programų modelį – taip vadinamas "semantines" programas. Tokių programų modelį užduoda pasirinktas skaičiavimo kelius ribojantis "kvotos predikatas" Q (quote predicate). Tokių predikatų pavyzdžiai yra:

- Kiekvienas kintamasis gali būti testuojamas daugiausiai k kartų.
- Daugiausiai s kintamųjų gali būti testuojami daugiau nei vieną kartą.

Tam, kad apibrėžti, kada programa P akceptuoja duotą vektorių $a \in \{0,1\}^n$, mes nagrinėjame tokį dviejų personų – Skaičiuotojo ir Trukdytojo – žaidimą. Tarkim programos P skaičiavimo kelias paiekė kokią nors jos viršūnę v, kurioje testuojamas kuris nors kintamasis x_i. Jeigu to testo atžvilgiu duota kvota Q dar neviršyta, tai Skaičiuotojas pratęsia skaičiavimo kelią iš viršūnės v išeinančiu lanku, pažymėtu skaičium a_i (kaip ir įprastose programose). Bet jeigu kvota jau viršyta, tai Trukdytojas gali pratęsti skaičiavima bet kuriuo iš dviejų viršūnę v paliekančių lankų. Skaičiuotojo tikslas yra pasiekti kurį nors lapą, pažymėtą skaičiumi 1 (tada jis "laimi"). Trukdytojo (corrupter) tikslas yra "trukdyti", tai yra, stengtis nuvesti Skaičiuotoją kuriuo nors klaidingu keliu. Tada laikome, kad

- $P(a) = 1$, jeigu Skaičiuotojas gali laimėti nepriklausomai nuo to, ką daro Trukdytojas.
- $P(a) = 0$, jei Skaičiuotojas negali laimėti net tuo atveju kai Trukdytojas yra "kooperatyvus" ir visada renkasi Skaičiuotojui labiausiai tinkamą kelią.

Pastebėkim, kad į prastos, ankstesniuose skyriuose nagrinėtos nedeterminuotos read-k programos ir $(1,+s)$-programos yra specialus semantinių programų atvejas (su atitinkamais kvotos predikatais): tuo atveju Trukdytojas paprasčiausiai neturi galimybės trukdyti, nes kvota niekada nėra viršijama. Taigi, mūsų įvestos semantinės programos apibendrina visus iki šiol nagrinėtus ribotų programų modelius ir duoda visą eilę naujų modelių – viskas priklauso nuo pasirinkto kvotos predikato.

Be kitų rezultatų, darbe (Jukna–Razborov 1992) mes išplėtėme mūsų įverčius (Teorema 5 bei Teorema 8) ir semantiniams tokių programu modeliams. Šitie rezultatai iki šiol lieka geriausi ir tuo pačiu nubrėžė mūsų dabartinių galimybių ribą – niekam iki šiol[11] nepavyko jų pagerinti.

10. Mažo gylio schemos

Darbe (Håstad, Jukna, Pudlák 1993) nagrinėjamas gylio-3 schemų sudėtingumas. Šios schemos yra dviejų tipų. Taip vadinamos Σ_3-schemos yra

[11] 2006 m. balandis

pavidalo

$$\bigvee_{i=1}^{s} \bigwedge_{j=1}^{r} \bigvee_{k \in S_{ij}} y_k;$$

čia y_k yra kintamasis x_i arba jo neigimas $\neg x_i$. Dualios Σ_3-schemoms yra taip vadinamos Π_3-schemos:

$$\bigwedge_{i=1}^{s} \bigvee_{j=1}^{r} \bigwedge_{k \in S_{ij}} y_k.$$

Tokių schemų sudėtingumu laikomas jose naudojamų loginių elementu $\wedge$ ir $\vee$ skaičius $s \cdot t$.

Gylio-3 schemos yra svarbus modelis, nes gavus aukštą (eksponentinį) apatinį įvertį tokioms schemoms mes galėtume tuo pačiu išpręsti vieną iš pagrindinių (ir intensyviai tiriamų) sudėtingumo teorijos problemų – įrodyti, kad konkreti Bulio funkcija su n kintamųjų negali būti realizuota logaritminio gylio loginėm schemom. Tam tikslui reikia įrodyti pavidalo $2^{\Omega(n)}$ apatinį įvertį gylio-3 schemoms. Deja, iki šiol žinomais metodais pavyksta gauti tik apatinius įverčius ne aukštesnius kaip $2^{\Omega(\sqrt{n})}$.

Tam, kad geriau suprasti, kokias Bulio funkcijas sunku realizuoti gylio-3 schemomis (ir kodėl), darbe (Håstad, Jukna, Pudlák 1993) mes pateikiam naują tokių schemų dydžio vertinimo metodą, paremtą taip vadinamos "baigtinės ribos" sąvoka. Tarp kitų rezultatų šiame darbe įrodyta, kad skirtumas tarp Σ_3 ir Π_3 schemų sudėtingumo gali būti net eksponentinis. Būtent, Bulio funkcija

$$f(x,y) = \bigvee_{i=1}^{\sqrt{n/2}} \bigwedge_{j=1}^{\sqrt{n/2}} (\neg x_{ij} \vee \neg y_{ij})$$

su n kintamųjų akivaizdžiai realizuojama Σ_3 schema su $O(n)$ elementų, Kita vertus, galioja tokia

Teorema 14 (Håstad–Jukna–Pudlák 1993)**.** *Bet kuri Π_3-schema realizuojanti funkciją f privalo turėti mažiausiai $2^{\Omega(\sqrt{n})}$ elementų.*

Kaip minėjome, darbe siūlomame metode mes naudojame vieną naują– taip vadinamų "baigtinių ribų" – sąvoką. Būtent, vektorius $b \notin A$ yra laikomas vektorių aibes $A \subseteq \{0,1\}^n$ *k-riba*, jeigu bet kurioms k koordinačių šis vektorius b sutampa visose šiose koordinatėse su bent vienu aibes A vektorium, t.y. vektorius b yra vektorių aibės A k-riba, jei

$$\forall S \subseteq \{1,\ldots,n\}\ |S| = k \implies \exists a \in A\ \forall i \in S:\ a_i = b_i.$$

Pagrindinė baigtinių ribų taikymo idėja yra tokia. Tegu $C(x)$ yra kokia nors k-CNF (conjunctive normal form)

$$C(x) = (\bigvee_{i \in S_1} x_i) \wedge (\bigvee_{i \in S_2} x_i) \wedge \cdots \wedge (\bigvee_{i \in S_r} x_i)$$

ir $A = \{a \in \{0,1\}^n : C(a) = 1\}$. Tarkim, $b \in \{0,1\}^n$ yra aibes A k-riba. Tada $C(b) = 1$. Kodėl? Pagal k-ribos apibrėžimą, visose koordinatese $i \in S_1$ vektorius b turi tas pačias reikšmes kaip ir kuris nors vektorius $a_1 \in A$; taigi, $b_i = 1$ kuriam nors $i \in S_1$. Argumentuojant tuo būdu toliau gauname, kad kievienoje aibėje S_j yra koordinatė $i \in S_j$ kuriai $b_j = 1$. Taigi, $C(b) = 1$.

Pagrindinę idėją, kaip baigtinės ribos naudojamos apatinių įverčių gavimui, iliustruoja tokia "ribinė lema" (limit lema). Kaip ir anksčiau, $|a|$ žymi vektoriaus a svorį, t.y. $|a| = |\{i : a_i = 1\}|$.

Lema 15 (Håstad–Jukna–Pudlák 1993)**.** *Tegu $A \subseteq \{0,1\}^n$ ir $|a| = s$ visiems $a \in A$. Jeigu $|A| \geq k^s + 1$, tai bent vienas vektorius $b \notin A$ yra aibės A k-riba.*

Taigi, bet kuri k-CNF $C(x)$, akzeptuojanti tik svorio s vektorius, negali akzeptuoti daugiau nei k^s tokių vektorių, nes priešingu atveju ji būtų priversta "daryti klaidą" (akzeptuoti vektorių, kurio svoris yra mažesnis nei s).

Ši (k-ribos) sąvoka vėliau pasirodė naudinga sprendžiant apatinių sudėtingumo įverčių gavimo problemą ir kitiems schemų modeliams. Įdomu pažymėti, kad naudojant baigtines ribas galima lengvai gauti tokius pačius (ir net kiek aukštesnius) apatinius įverčius gylio-3 schemom, kaip ir įverčiai gaunami taip vadinamu "atsitiktinių projekcijų" metodu, taikomu visuose kituose riboto gylio schemų sudėtingumą tiriančiuose darbuose.

11. Schemos su slenkstiniais elementais

Darbe (Jukna 1995a) nagrinėjamos gylio-3 schemos, kuriose vietoj konjunkcijos ($\wedge$) ir disjunkcijos ($\vee$) operacijų leidžiama naudoti bendresnes operacijas, būtent bet kokias "slenkstines" funkcijas (threshold functions).

Slenksčio-r funkcija yra Bulio funkcija T_r^m tokia, kad

$$T_r^m(x_1 \ldots, x_m) = 1 \iff x_1 + \cdots + x_m \geq r.$$

Schemos su tokiais elementais buvo (ir yra) plačiai tiriamos, kadangi jų pagalba galima modeliuoti neuroninius tinklus. Be to, tokios schemos yra daug galingesnės, nei $\{\wedge, \vee\}$-schemos: jau net gylio-3 schemomis galima ekonomiškai realizuoti daugelį pagrindinių operacijų, kaip dviejų skaičių dalyba ir pan. Todėl nenuostabu, kad apatinių įverčių gavimo problema tokioms (net gylio-3) schemoms yra sunkesnė. Pastebekime, kad šios funkcijos (dizjunkcija $\vee$ ir konjunkcija $\wedge$) yra tik labai specialus slenkstinių funkcijų atvejas:

$$\bigvee_{i=1}^{m} x_i = T_1^m(x_1 \ldots, x_m) \quad \text{ir} \quad \bigwedge_{i=1}^{m} x_i = T_m^m(x_1 \ldots, x_m).$$

Iki šiol pavyko įrodyti tik kelis eksponentinius įverčius slenkstinėms gylio-3 schemoms su papildomais apribojomais funkcijoms, naudojamoms pirmame (artimiausiam kintamiesiems) lygyje:

(1) Jei pirmam lygyje naudojamos funkcijos priklauso tik nuo $c \log n$ ($c < 1$) kintamųjų, tai bet kuri gylio-3 slenkstinė schema funkcijai

$$f_n(x) = \bigoplus_{i=1}^{n} \bigwedge_{j=1}^{\log n} x_{ij}$$

privalo turėti mažiausiai $\exp(\Omega(n^{\epsilon}))$ elementų (Babai–Nissan–Szegedy 1992; Håstad–Goldmann 1990).

(2) Jei pirmam lygyje naudojamos funkcijos yra $\wedge$-funkcijos arba bet kokios Bulio funkcijos, priklausančios tik nuo $n^{1-\epsilon}$ kintamųjų, tai bet kuri gylio-3 slenkstinė schema funkcijai

$$g_n(x) = \bigoplus_{i=1}^{n} \bigwedge_{j=1}^{\log n} \bigoplus_{l=1}^{n} x_{ijl}$$

privalo turėti mažiausiai $n^{\Omega(\log n)}$ elementų (Razborov–Wigderson, 1993).

Šitų rezultatų įrodymui naudojamos funkcijos $f_n(x)$ ir $g_n(x)$ yra parinktos taip, kad jos naudoja operaciją – sumavimą modulo 2 – kuri tam tikra prasme yra "priešiška" monotoninei slenkstinių funkcijų prigimčiai:

$$T_r^m(x) = \bigvee_{S \subseteq [m]: |S| = r} \bigwedge_{i \in S} x_i.$$

Ilgai nebuvo jokio eksponentinio įverčio funkcijai, kuri būtų monotoninė.

Darbe (Jukna 1995a) mes įrodome tokį įvertį. Būtent, mes nagrinėjame slenkstinės funkcijos T_r^n realizavimo gylio-3 schemomis, kurių elementai yra slenkstinės funkcijos T_s^n ir T_{n-s}^n su $s \leq r$. Tokias schemas vadinsime s-schemomis.

Funkciją T_r^n galima realizuoti trivialia gylio-2 1-schema – būtent, monotonine DNF – susidedančia iš $\binom{n}{r}+1 \leq e^{r(1+\ln(n/r))}$ elementų. Khasin (1969) įrodė, kad šią funkciją galima realizuoti daug mažesne schema, jeigu vietoj gylio-2 leisime naudoti gylio-3 schemas. Būtent, jis įrodė, kad T_r^n galima realizuoti gylio-3 1-schema turinčia tik $e^r n \ln n$ elementų, net jeigu kiekvienas elementas pirmame (arčiausiame prie įėjimų) schemos lygyje skaičiuoja funkciją priklausanįą nuo n/r kintamųjų. Norimą schemą Khasin gavo imdamas dizjunkziją susdedančią iš e^r atsitiktinių tokių gylio-2 schemų kopijų

$$\Big(\bigvee_{i \in S_1} x_i\Big) \wedge \Big(\bigvee_{i \in S_2} x_i\Big) \wedge \cdots \wedge \Big(\bigvee_{i \in S_r} x_i\Big)$$

kur $S_1, \ldots, S_r$ yra atsitiktinis aibės $\{1, \ldots, n\}$ suskaidymas į r dydžio n/r dalių. Kai r yra palygint mažas, būten kai $r \leq \ln n$, Radhakrishnan (1994) pagerino šį viršutinį įvertį iki $e^{\sqrt{r} \ln r} n \ln n$.

Teorema 16 (Jukna 1995a)**.** *Tegu $s = o(r^{\epsilon/2})$ kuriai nors konstantai $\epsilon > 0$. Jei pirmam lygyje naudojamos funkcijos yra $\wedge$-funkcijos arba bet kokios*

Bulio funkcijos, priklausančios tik nuo $n^{1-\epsilon}$ kintamųjų, tai bet kuri gylio-3 slenkstinė s-schema funkcijai T_r^n privalo turėti mažiausiai $n^{\Omega(r/s)}$ elementų.

Be kita ko ši teorema rodo, kad (parametrams $r > \ln n$) Khashin'o konstrukcija yra iš esmės optimali. Teoremos įrodyme mes vėlgi naudojame baigtinės ribos sąvoką, būtent – Lemą 15.

12. Monotoninės schemos

Jei Bulio funkcija f yra monotonine – t.y. jei pakeitus bet kurią vieną vectoriaus koodinatę iš 0 į 1 funkcijos reikšmė negali pasikeisti priešinga kryptimi (iš 1 į 0) – tai tokią funkciją galima realizuoti monotonine shema, t.y. shema nenaudojančia neigimo operacijos ($\neg$). Bet netgi šitame ribotame modelyje aukščiausias apatinis įvertis ilgą laika (iki 1985) buvo tik $4n$.

Esminį šuolį šioje kryptyje padarė rusų matematikas A. Razborovas 1985 metais: jis įrodė, kad Bulio funkcija Clique, atpažįstanti ar duotas grafas turi kliką (pilną pografį) su $\sqrt{n}$ viršūnių, reikalauja $n^{c \log n}$ monotoninių operacijų. Už šį rezultatą jam 1990 metais buvo skirta Nevanlinna premija. Vis dėlto, pati problema tuom dar nebuvo pilnai išspręsta: liko neaišku kaip gauti aukštus įverčius kitoms funkcijoms (Razborovo įrodymas buvo paremtas labai specifinėm pačios funkcijos Clique savybėmis). Todėl pats Razborovas 1986 metų Matematikų Kongrese (Berkley) iškėlė tokią problemą:

> Ar egzistuoja bendra ir lengvai patikrinama Bulio funkcijų kombinatorinė savybė, daranti šias funkcijas sunkiai realizuojamom nenaudojant neigimo operacijos?

Ši problema buvo išspręsta darbuose (Jukna 1997,1999): jeigu duotos Bulio funkcijos f negalima aproksimuoti polinominio dydžio konjunktyvinėm ir dizjunktyvinėm formom, tai šios funkcijos negalima realizuoti polinominio dydžio monotonine shema. Įdomu (ir gana netikėta), kad šis kriterijus galioja ir daug platesneje loginių shemomų klaseje, kur kartu su monotoninėm dvejetainėm operacijom $x \vee y$ ir $x \wedge y$ galima naudoti bet kurias monotonines *realias* operacijas $g(x, y)$.

Tam, kad suformuluoti mūsų kriterijų, priminkime, kad ilgio ℓ k-DNF (disjunctive normal form) yra Bulio funkcija $g(x)$ su kintamaisiais $x = (x_1, \ldots, x_n)$ kurią galima išreikšti pavidalu

$$C(x) = \bigvee_{i=1}^{\ell} \bigwedge_{j \in S_i} x_j$$

kur $|S_i| = k$ visiems $i = 1, \ldots, \ell$. Ilgio ℓ k-CNF (conjunctive normal form) yra apibrežta dualiai:

$$D(x) = \bigwedge_{i=1}^{\ell} \bigvee_{j \in S_i} x_j.$$

Toliau (kaip įprasta) rašome $f \leq g$, jei $f(a) \leq g(a)$ visiems $a \in \{0,1\}^n$.

Tegu $f : \{0,1\}^n \to \{0,1\}$ yra monotoninė Bulio funkcija. Sakykime, kad f yra *t-paprasta*, jei visiems skaičiams $2 \leq s, r \leq n$ egzistuoja ilgio tr^s s-CNF C ir ilgio ts^r r-DNF D bei aibė $I \subseteq \{1, \ldots, n\}$ su $|I| \leq s-1$ elementų tokie, kad galioja bent viena iš šių dvieju nelygybių:

$$C \leq f \quad \text{arba} \quad f \leq D \vee \Big(\bigvee_{i \in I} x_i \Big).$$

Funkcija yra *t-sudėtinga*, jei ji nėra t-paprasta.

Darbe (Jukna 1999) įrodytas toks bendras apatinių įverčių monotoninėms schemoms gavimo kriterijus, išsprendęs anksčiau minėtą Razborov'o problemą.

Teorema 17 (Jukna 1999)**.** *Kiekviena monotoninė schema, skaičiuojanti t-sudėtingą monotoninę funkcija, privalo turėti mažiausiai t elementų.*

Šio kriterijaus įrodyme mes vėl naudojame jau minėtą "baigtinės ribos" sąvoką. Naudojant šį kriterijų, darbe (Jukna 1999) buvo įrodyti eksponentinai apatiniai įverčiai tokiom Bulio funkcijom, kurioms originalus Razborov'o metodas nedirbo.

13. Optimalių schemų nestabilumas ir jo taikymai

Tegu S yra kokia nors loginė schema (kurioje nors bazėje), realizuojanti kurią nors Bulio funciją $f_S : \{0,1\}^n \to \{0,1\}$, ir tegu $|S|$ yra toje schemoje naudojamų loginių elementų skaičius. Schema S yra *optimali*, jei funkcijos f_S negalima realizuoti naudojant mažiau negu $|S|$ elementų. Taigi, optimalios schemos yra "nestabilios" elementų pašalinimo atžvilgiu: išmetus iš S bet kurį vieną elementą, gauta schema privalo "daryti klaidą", t.y. realizuoti kokią nors Bulio funkciją $g \neq f_S$.

Darbe (Jukna 1991) pastebėta, kad optimalio schemos yra nestabilios ne tik elementų pašalinimo atžvilgiu. Pasirodo, kad tokia schema privalo daryti klaidą, t.y. realizuoti kokią Bulio funkciją $g \neq f_S$, ne tik tada, kai pašalinsime kurį nors jos elementą – ji privalo daryti klaidą net tada, kai koks nors jos elementas pakeičiamas kitu elementu; bendras elementų skaičius tokiu atveju lieka tas pats!

Kiekvienas schemos elementas yra kokia nors Bulio funkcija $e(x,y)$ su dviem kintamaisiais, tarkim dizjunkcija $e(x,y) = x \vee y$ ar konjunkcija $e(x,y) = x \wedge y$. Darbe (Jukna 1991) su kiekvienu tokiu elementu e susiejama atitinkama elementu aibė $H(e)$. Pavydžiui, $H(\vee)$ susideda iš visų funkcijų $h(x,y)$, išskyrus $h(x,y) = x \vee y$ ir $h(x,y) = x \oplus y$; taigi $|H(\vee)| = 14$.

Teorema 18 (Jukna 1991)**.** *Tegu S yra optimali loginė schema ir e kuris nors jos elementas. Tegu $S[e \to h]$ yra schema gauta pakeitus schemoje S elementą e kuriuo nors elementu $h \in H(e)$. Tada $f_{S[e \to h]} \neq f_S$.*

Darbe (Jukna 1991) šios teoremos pagrindu gautas optimalių schemų nestabilumu paremtas apatinių įverčių metodas. Čia mes tik trumpai pateiksime pagrindinę jo idėją. Pagal Teoremą 18, kiekvieno optimalios schemos S elemento e "būtinumą" turi "paliūdyti" bent vienas vektorius a toks, kad $f_{S[e\to h]}(a) \neq f_S(a)$. Naudojant tuos "liūdininkus" a, bandoma gauti papildomos informacijos apie schemos S *struktūrą*. Pagaliau, naudojant šią papildomą informaciją, bandoma įrodyti, kad schema S turi turėti daug elementų.

Tame pačiame darbe šis metodas yra demonstruojamas keletu naujų apatinių įverčių loginėms schemoms be taip vadinamų "nulinių šakų".

14. Loginės įrodymos sistemos: rezoliuzija

Rezoliucija yra viena iš plačiausiai naudojamų loginių įrodymų sistemų. Ši sistema plačiai naudojama ir praktikoje: loginis programavimas, duomenų bazės ir pan.

Pačią rezoliuciją pasiūlė Blake dar 1937 metais, bet ji ilgą laiką nebuvo naudojama kol Davis–Putnam (1960) ir Robinson (1965) išpopuliarino ją kaip teiginių logikos teoremų įrodymo sistemą.

Tegu $x_1, \ldots, x_n$ būna Bulio kintamieji. Priminkime, kad *literalas* yra kintamasis x_i arba jo neigimas $\neg x_i$. *Monomas* yra kokių nors literalų konjunkcija, o *klausas* (clause) yra kokių nors literalų disjunkcija. Paprastai rezoliucijos sistema dirba su klausais. Patogumo dėlei mes naudosime ekvivalentų apibrėžimą, naudojantį monomus vietoje klausų.

Tegu $F = \{M_1, M_2, \ldots, M_t\}$ būna kokia nors DNF (t.y. monomų aibė). Tokia DNF vadinama *tautologija*, jei kiekvienam vektoriui $a \in \{0,1\}^n$ egzistuoja i toks, kad $M_i(a) = 1$. Rezoliucijos tikslas yra, gavus bet kokią tautologiją, įrodyti, kad ji iš tikro yra tautologija. Tą ji daro grynai mechaniškai pradėdama darba su visa monomų aibe $F = \{M_1, M_2, \ldots, M_t\}$ ir bandydama "pagimdyti" (sukurti) vis naujus monomus pagal tokią taisyklę

$$\frac{M \wedge x_i \qquad N \wedge \bar{x}_i}{M \wedge N} \tag{19}$$

vadinamą *rezoliucijos taisykle*, kurios prasmė yra tokia: Turint monomus $M \wedge x_i$ ir $N \wedge \bar{x}_i$ galima pagimdyti naują monomą $M \wedge N$. Tikslas yra pagimdyti "tuščią" monoma $\Lambda = \emptyset$, kuris interpretuojamas kaip visur teisingas monomas, t.y. $\Lambda(a) = 1$ visiems $a \in \{0,1\}^n$. Tas "gimdymo" procesas vadinamas tautologijos F (rezoliuciniu) įrodymu. Nesunku įsitikinti, kad DNF F yra tautologija tada ir tik tada, kai taikant rezoliucijos taisyklę iš aibes F galima pagimdyti tuščią monomą. Pastebėkime, kad viena ir ta pati tautologija gali turėti daug skirtingų įrodymų – viskas priklauso nuo to, kokia eile resoliucijos taisykle yra taikoma.

Tautologijos F išvedimo sudėtingumas $RES(F)$ yra minimalus jos įrodymui naudojamas resoliucijos taisyklės (19) taikymų skaičius. Tikslas yra

įrodyti, kad kuri nors konkreti tautologija F reikalauja eksponentinio skaiciaus $2^{\Omega(|F|)}$ resoliucijos taisyklės taikymų. Plačiausiai tiriama buvo tautologija formalizuojanti gerai žinomą Dirichlet prinzipą:

> *Dirichlet prinzipas*: Jei $m \geq n+1$ balandžių bando sutūpti į n narvelių ir kiekviename nervelyje telpa tik vienas balandis, tai bent vienas balandis turi likti be savo narvelio.

Tam kad apibrėžti šį principą atitinkačią tautologiją, imame nm Bulio kintamųjų $x_{i,j}$ tokių, kad

$$x_{i,j} = 1 \iff i\text{-tasis balandis tupi } j\text{-tajam narvelyje.}$$

Atitinkama tautologija PHP_n^m $(m \geq n+1)$ susideda iš m monomų

$$\neg x_{i,1} \wedge \neg x_{i,2} \wedge \cdots \wedge \neg x_{i,n}, \qquad i = 1, \ldots, m$$

sakančių, kad i-tasis balandis neturi narvelio, ir iš $n\binom{m}{2}$ monomų

$$x_{i,k} \wedge x_{j,k} \qquad 1 \leq i < j \leq m, 1 \leq k \leq n$$

sakančių, kad du balandžiai i ir j sėdi narvelyje k.

Pirmą fundamentalų rezultatą apie rezoliucijos sudėtingumą įrodė Haken 1985 metais:

Teorema 20 (Haken 1985)**.** $RES(PHP_n^{n+1}) = 2^{\Omega(n)}$.

Po to buvo bandoma išplėsti šį rezultatą dviem kryptim:

(i) Leisti daugiau balandžių, t.y. nagrinėti atvejį, kai turime $m > n+1$ alandžių – tuo atveju pats principas darosi vis "teisingesnis" ir, tuo pačiu, jo įrodymas gali (bent jau potencialiai) būti daug trumpesnis.
(ii) Leisti įrodyme naudoti bendresnes taisykles nei (19).

Darbe (Jukna 1998a) mes gavome rezultatus abiem šiom kryptim. Mes nagrinėjame taip vadinamą *semantinę rezoliuciją*. Skirtumas nuo klasikinės rezoliucijos yra tas, kad dabar vietoj taisyklės (19) naujų monomų gimdymui mes leidžiame naudoti bet kokias taisykles pavidalo

$$\frac{M_1,\ M_2,\ \ldots,\ M_d}{M}. \tag{21}$$

Vienintelė sąlyga monomans $M, M_1, \ldots, M_d$ yra, kad $M \leq M_1 \vee M_2 \vee \cdots \vee M_d$, t.y. jei kuris nors vektorius $a \in \{0,1\}^n$ išpildo monomą M $(M(a) = 1)$, tai jis turi išpildyti bent vieną iš monomų $M_1, \ldots, M_d$. Pastebėkime, kad klasikinė resoliucijos taisykle (19) yra tik specialus šios bendros taisyklės atvejas kai $d = 2$.

Atitinkamam sudėtingumo matui $RES_d(F)$ šioje žymiai platesnėje loginėje sistemoje mes įrodėme toki rezultatą:

Teorema 22 (Jukna 1998a)**.** $RES_d(PHP_n^m) = 2^{\Omega(n^2/md)}$.

Panašūs eksponentiniai įverčiai gauti ir visai eilei kitų tautologijų. Visi šie rezultatai tiesiogiai seka iš vieno bendro, darbe (Jukna 1998a) įrodyto apatinio įverčio matui $RES_d(F)$, galiojančio visoms tautologijoms, paremtoms taip vadinamu "blokuojančių aibių prinzipu". Dirichlet principas yra tik atskiras to bendresnio principo atvejas. Šios (bendros) theoremos formulavimui reikėtų visos eilės baigtinių aibių kombinatorikos sąvokų, tad čia mes jos neformuluosime.

15. Kommunikacinis sudėtingumas

Komunikaciniai protokolai yra vienas iš fundamentalių skaičiavimo modelių.[12] Be akivaizdžių taikymų kasdieniniame gyvenime, jie yra svarbūs ir algoritmų sudėtingumo teorijoje: jie leidžia modeliuoti informacijos tekėjimą tarp skirtingų algoritmo dalių.

Komunikacinis sudėtingumas yra svarbus *informacijos kiekio* funkcijoje matas. Šio mato esmę galima apibūdinti tarkim tokiu pavyzdžiu. Turime du žaidėjus, kurie gyvena skirtinguose miestuose. Vienas žaidėjas gauna pirmini skaičių x, antras gauna sudėtini skaičių y; čia $x, y \leq 2^n$. Žaidėjų tikslas yra surasti pirminį skaičių $p < 2n$ tokį, kad $x \not\equiv y \pmod{p}$. (Toks skaičius p visada egzistuoja.) Klausimas yra, kiek informacijos bitų žaidejai turi siūsti pirmyn ir atgal blogiausiu atveju (t.y. kuriai nors skaičių porai x, y) kol jie tokį skaičių p suras? Šis maksimalus siūstų bitų skaičius $t(n)$ ir yra duotos problemos *komunikacinis sudėtingumas*. Aišku, kad $t(n) \leq n + \log n$: naudodamas n bitu pirmas žaidejas persiunčia antram visą savo skaičių x ir antras žaidejas (turėdamas abu skaičius x ir y) suranda reikiama pirmini skaičių p ir persiunčia pirmam žaidėjui skaičiaus p ($\log n$ ilgio) kodą. Tačiau iki šiol nėra žinoma koks yra tikras šios problemos komunikacinis sudėtingumas: žinoma tik, kad žaidėjai privalo siūsti mažiausiai $c \log n$ bitų

15.1. **Dviejų žaidėjų komunikacija.** Dviejų žaidėjų komunikacijos modelį pasiūlė Abelson (1978) ir Yao (1979). Turime Bulio funkciją $f(X)$ su n kintamųjų $X = \{x_1, \ldots, x_n\}$ ir aibės X suskaidymą $\pi = (Y, Z)$ į dvi lygias dalis. Du žaidėjai – paprastai vadinami Alice ir Bob – nori (kooperatyviai) skaičiuoti funkciją $f(X)$. Tačiau Alice mato tik kintamųjų Y reikšmes, o Bob tik kintamųjų Z reikšmes. (Pati funkcija f yra žinoma abiems žaidėjams.) Turėdami šitą dalinę informaciją apie visų kintamųjų reikšmes, žaidėjai bando nustatyti funkcijos reikšme komunikuodami (pagal išanksto susitartą protokolą) vienas kitam tam tikrą informacija. Komunikacinis funkcijos $f(X)$ sudėtingumas $\mathrm{c}(f, \pi)$ kintamųjų X suskaidymo π atžvilgiu yra apibrėžiamas kaip maksimalus komunikuotų bitų skaičius imant geriausią komunikavimo protokolą. (Absoliutus) *komunikacinis funkcijos* $f(X)$ *sudėtingumas* $\mathrm{c}(f)$ yra skaičiaus $\mathrm{c}(f, \pi)$ minimumas pagal visus galimus kintamųjų X suskaidymus π. T.y. kiekvienai funkcijai f leidžiama pasirinkti

[12] Daugiau informacijos apie šį įdomų modelį galima rasti knygoje: E. Kushilevitz, N. Nisan, *Communication Complexity*, Cambridge University Press, Cambridge, 1997.

geriausią (labiausiai tai funkcijai tinkamą) kintamųjų suskaidymą. Nagrinėjami ir *nedeterministiniai* komunikacijos protokolai (kur žaidėjams leidžiama "spėlioti ir tikrinti"); atitinkamas sudėtingumo matas tuo atveju žymimas $\mathrm{nc}(f)$. Pastebėkime, kad $\mathrm{nc}(f)$ kaip ir $\mathrm{c}(f)$ niekada neviršija kintamųjų skaičiaus n.

Pirmą įvertį $\mathrm{nc}(f) = \Omega(n)$ įrodė Yao 1981 metais vienai specialiai funkcijai f, koduojančiai grafų izomorfizmo problema. Tame pačiame darbe jis iškėlė problemą gauti tokius įverčius kitoms funkcijoms pažymėdamas, kad "matomai tai yra sunki problema". Sunkumas čia glūdi tame, kad suskaidymas π nėra fiksuotas – jei π fiksuotas, tai apatinių įverčių problema matui $\mathrm{nc}(f, \pi)$ yra žymiai lengvesnė. Ir iš tikrųjų, Yao "pranašavimas" buvo teisingas: per ilgą pavyko gauti tik keletą naujų iverčių funkcijoms, koduojančioms tam tikras grafų sąvybes: Ja'Ja' (1984), Papadimitriou ir Sipser (1984), Hajnal, Maass ir Turán (1988), Meinel and Waack (1992) ir keletas kitų autorių. Visi šie įrodymai buvo paremti paprasta idėja: transformuoti problemą į *fiksuoto* suskaidymo π atvejį, kur apatiniai įverčiai lengvai gaunami. Deja, už tą "paprastumą" reikia mokėti: pačioje transfomacijoje autoriai buvo priversti naudoti gilius kombinatorikos rezultatus, tokius kaip Ramsėjaus tipo rezultatai, Szemerédi Regularumo Lemą grafams ir kt.

Nepriklausomai nuo tų rezultatų (bei jų nežinant) ir su kitu tikslu (būtent, gauti apatinius įverčius alternuojančiom kontaktinėm schemom) darbe (Jukna 1987) buvo pasiūlytas visiškai kitas ir daug paprastesnis apatinių įverčių matui $\mathrm{nc}(f)$ gavimo metodas. Deja, vakaruose šis metodas ilgą laiką buvo nežinomas.

Kiekvienas kintamųjų aibės X suskaidymas $\pi = (Y, Z)$ duoda mums 0-1 matricą (taip vadinamą funkcijos f "komunikacinę matricą) $A_{f,\pi} = \{f(u, v)\}$ su $u \in \{0,1\}^Y$ ir $v \in \{0,1\}^Z$. Mus domina tokie 0-1 matricų A kombinatoriniai parametrai:

- $tr(A)$ = maksimalus skaičius t toks, kad A turi t nenulinių elementų, jokie du iš kurių guli toj pačioj matricos A "linijoje" (eilutėje ar stulpelyje);
- $w(A)$ = maksimalus vienetų skaičius kurioj nors matricos A linijoje;
- $cl(A)$ = maksimalus skaičius k toks, kad matrica A turi bent vieną $k \times k$ minorą, susidedanti vien iš vienetų.

Tada apibrėžiame tokį Bulio funkcijų f matą

$$D(f) = \min_{\pi} \frac{|A_{f,\pi}|}{w(A_{f,\pi}) \cdot cl(A_{f,\pi})}.$$

Teorema 23 (Jukna 1987)**.** *Kiekvienai Bulio funkcijai f galioja*

$$\mathrm{nc}(f) \geq \log_2 D(f).$$

Teoremos įrodyme mes naudojame klasikinį König-Egervary rezultatą, $tr(A)$ sutampa su mažiausiu visus matricos A nenulinius elementus padengiančių linijų skaičiumi. Kadangi matą $D(f)$ daugeliu atveju yra nesunku

įverinti, ši teorema leidžia įrodyti aukštus komunikacinio sudėtingumo įverčius visai eilei Bulio funkcijų.

15.2. k **žaidėjų komunikacija.** Taikymuose dažnai tenka turėti reikalo su daugiau nei dviem žaidėjais. Todėl buvo pasiūlyti įvairūs komunikacinių protokolų praplėtimai ir šiai (sunkesnei) situacijai.

Ypač įdomus (ir svarbus taikymuose) yra taip vadinamas "skaičių ant kaktos" (numbers on the forehead) komunikacinis modelis. Čia mes turime $k \geq 2$ žaidėjų, kurių tikslas yra skaičiuoti duotą funkciją $f(x_1, \ldots, x_k)$; pati funkcija yra žinoma visiems žaidėjams. Vienintelis apribojimas yra tas, kad i-tasis žaidejas negali matyti i-tojo argumento x_i – visi kiti argumentai jam yra matomi. T.y. i-tasis žaidejas mato tik dalinį vektorių $(x_1, \ldots, x_{i-1}, *, x_{i+1} \ldots, x_k)$. Galima įsivaizduoti, kad žaidejai sėdi prie apvalaus stalo ir i-tasis argumentas x_i yra "užrašytas" i-tajam žaidejui ant kaktos.

Gavę vektorių $x = (x_1, \ldots, x_k)$ žaidėjai stengiasi (kooperatyviai) nustatyti funkcijos reikšmę $f(x)$. Tam tikslui jiems leidžiama tarp savęs komunikuoti, t.y. kuris nors žaidėjas užrašo ant lapo kokią nors informacija apie jam matoma vektoriaus x dalį, po to kuris nors kitas žaidejas, matydamas tą informaciją užrašo ant to paties lapo kokią nors informacija apie jam matoma vektoriaus x dalį, ir tt. *Komunikacinis* funkcijos $f(x)$ *sudėtingumas* $C_k(f)$ yra mažiausias funkcijos $f(x)$ visoms reikšmėms nustatyti reikalingo lapo dydis. (Lapas yra "nutrinamas", taip kad tą patį lapa galima naudoti skirtingiems vektoriams x.)

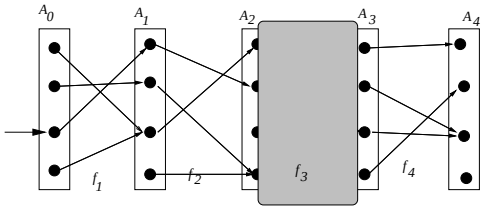
Jei visi skaiciai x_i yra ne didesni nei $N = 2^n$, tai $C_k(f) \leq n$: pirmas žaidėjas užrašo ant lapo visą skaičių x_1; po to antras žaidėjas žino visą vektorių x ir gali pranešti funkcijos reikšmę. Tačiau kai kurioms funkcijom užtenka žymiai mažesnio komunikavimo. Imkim, pavyzdžiui, funkciją $f(x_1, x_2, x_3) = 1 \iff x_1 + x_2 + x_3 = N$. Naudodami kai kuriuos netrivialius Ramsėjaus-tipo kombinatorinius rezultatus apie aritmetines progresijas, Chandra, Furst and Lipton (1983) įrodė, kad šiuo atveju galioja $C_3(f) = O(\sqrt{n})$. Taigi, komunikaciniai protokolai gali naudoti netrivialius matematinius rezultatus ir todėl yra nenuostabu, kad apatinių įverčių įrodymas tokiems protokolams yra netrivialus dalykas. Pagrindinis sunkumas čia glūdi tame, kad žaidėjų matoma informacija "kertasi" ir (bent iš principo) jie gali tą bendrą informaciją naudoti nerašydami nieko ant lapo.

Geriausias iki šiol pasiektas rezultatas buvo apatinis įvertis $C_k(f) = \Omega(n/2^k)$. Jį įrodė Babai, Nisan ir Szegedy 1992 metais. Tačiau šio rezultato silpnumas glūdi tame, kad jis darosi trivialus, kai žaidėjų skaičius k yra didesnis nei $\log n$. Kita vertus, mums reikia apatinių įverčių protokolams su $k > \log n$ žaidėjų, nes (kaip pastebėjo Hastad ir Goldmann 1991), tai leistų kaip išvadą gauti eksponentinius įverčius algebrinėms kontantinio gylio schemoms – ši pastaroji problema yra tarp aktualiausių atvirų

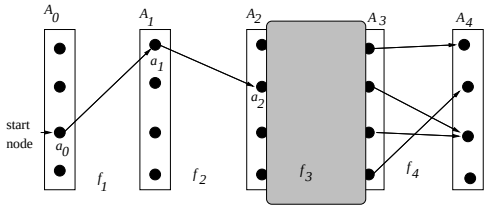
sudėtingumo teorijos problemų. Įdomu, kad tai išvadai gauti pakaktų pagerinti Babai, Nisan ir Szegedy apatinę ribą $\Omega(n/2^k)$ bent ribotiems, taip vadinamiems *vienakrypčiams* (one-way) protokolams: pirma šneka pirmas žaidėjas, po to antras ir taip toliau; paskutinis žaidėjas praneša rezultatą. Tačiau net tokiems ribotiems protokolams niekam iki šiol nepavyko gauti geresnio nei $\Omega(n/2^k)$ įverčio.

Tam, kad geriau suprasti žaidėjų skaičiaus įtaką komunikaciniam sudėtingumui, darbuose (Damm, Jukna 1995) ir (Damm, Jukna, Sgal 1996) mes nagrinėjame taip vadinamus "konservatyvius" komunikacijos protokolus. Tai yra vienakrypčiai protokolai su tokiu apribojimu: vietoj dalinio vektoriaus $(a_1, \ldots, a_{i-1}, *, a_{i+1} \ldots, a_k)$ i-tasis žaidejas mato $(a_{i+1}, \ldots, a_k)$ ir žino funkciją $f(a_1, \ldots, a_{i-1}, x_i, \ldots, x_k)$. Tai yra, jis nemato pačių skaičų $a_1, \ldots, a_{i-1}$, o tik žino kokią tie skaičiai įtaką turi funkcijos f reikšmei.

Konkreti darbe tiriama funkcija yra gerai žinoma "pointer jumping" funkcija $\mathrm{Jump}_{n,k}$. Ši funkcija yra svarbi, nes ji modeliuoja daugelį įvairiuose taikymuose iškylančių funkcijų, kaip permutacija, adresavimo funkcija, konvoliucija, dvejetainių skaičių daugyba ir kt.



Piešinys 5. Funkcija Jump su $m = 4$ ir $k = 4$. Uždengta dalis reiškia, kad trečias žaidejas jos nemato. Visa kita jam matoma.



Piešinys 6. Konservatyvus protokolas

Pati ši funkcija $\mathrm{Jump}_{n,k}(x_1, \ldots, x_k)$ apibrėžiama šitaip. Kiekvienas argumentas x_i interpretuojamas kaip funkcija [13] $x_i : [m] \to [m]$. Tada

$$\mathrm{Jump}_{n,k}(x_1, \ldots, x_k) = x_k(x_{k-1}(\ldots x_2(x_2(1)) \ldots)))$$

[13] Kur kaip įprasta $[m] = \{1, \ldots, m\}$.

Tai yra, pradedant nuo taško $1 \in [m]$ mes iš eilės taikom funkcijas $x_1, \ldots, x_k$ (funkcijų superpozicija) kol gaunam kažkokį taška $y \in [m]$; šis taškas ir yra funkcijos reikšmė. Pastebėkime, kad kiekvieno x_i dvejetainis ilgis yra $n = \log m^m = m \log m$ (mes turime m^m funkcijų iš $[m]$ į $[m]$). Jei funkcijas $x_i : [m] \to [m]$ mes interpretuosime kaip dvidalius grafus, tai funkciją $\mathrm{Jump}_{n,k}$ galima pailiustruoti kaip pavaizduota piešinyje 5. Apribojimas "konservatyvus" iliustruojamas piešinyje 6.

Mūsų pagrindiniai rezultatai konservativiems vienpusiams funkcijos $\mathrm{Jump}_{n,k}$ komunikaciniams protokolams su k žaidėjų yra tokie: [14]

(1) Apatinis įvertis $\Omega(n/k^2)$ kai $k = O(n^{1/3-\epsilon})$.
(2) Beveik tikslus įvertis $\Theta(n \log^{k-1} n)$ kai $k \leq \log^* n$.

Šis rezltatas žymiai pagerino Babai, Nisan ir Szegedy apatinę ribą $\Omega(n/2^k)$ ir iki šiol[15] lieka vieninteliu rezultatu k-žaidėjų komunikacinio sudėtingumo srityje, galiojančiu $k \gg \log n$ žaidėjams. Įdomi atvira problema yra išplėsti mūsų rezultatą nekonservatyviems protokolams.

Santraukoje cituota literatūra

[1] L. Babai, N. Nisan, and M. Szegedy, (1992): Multiparty protocols, pseudorandom generators for logspace, and time-space trade-offs, *J. Comput. System Sci.* **45**, pp. 204–232.

[2] A. Blacke, (1937): *Canonical expressions in Boolean algebra*, PhD thesis, University of Chicago, 1937.

[3] S. Buss and G. Turán, (1988): Resolution proofs of generalized pigeonhole principles, *Theor. Comput. Sci.* **62**, pp. 311–317.

[4] A. K. Chandra, M. L. Furst, and R. J. Lipton, (1983): Multi-party protocols. In *Proc. 15th STOC* , pp. 94–99.

[5] C. Damm, S. Jukna and J. Sgall, (1996): Some bounds on multiparty communication complexity of pointer jumping. In *Proc. 13th Ann. Symp. on Theor. Aspects of Comput. Sci.*, Springer Lecture Notes in Comput. Sci., **1046** (1996), pp. 643–654. Journal version in: *Computational Complexity* **7**:2 (1998), pp. 109–127.

[6] M. Davis and H. Putnam, (1960): A computing procedure for quantification theory, *J. of the ACM* **7**(3), pp. 210–215.

[7] A. Ehrenfeucht and D. Haussler, (1989): Learning decision trees from random examples, *Information and Computation* **82**, pp. 231-246.

[8] A. Haken, (1995): Counting Bottlenecks to Show Monotone P≠NP. In: *Proc. of 36th FOCS,* (1995), 36–40.

[14] $\log^i n = \log \log \ldots \log n$ (i kartų) yra iteruoto logaritmo funkcija ir $\log^* n$ yra mažiausias skaičius i, kuriam $\log^i n < 1$.

[15] 2006 metų balandis

[9] J. Hastad, S Jukna and P. Pudlák, (1993): Top-Down Lower Bounds for Depth-Three Circuits. In: *Proc. of 34th FOCS*. Journal version in: *Computational Complexity*, 5 (1995), pp. 99-112.

[10] S. Jukna, (1984a): Convolutional characterization of computabilty and complexity of computations. In: *Colloquia Mathematica Societatis János Bolyai* **44**, pp. 251–270.

[11] S. Jukna, (1984b): Succinct data representations and the complexity of computations. In: *Colloquia Mathematica Societatis János Bolyai* **44**, pp. 271–282.

[12] S. Jukna, (1985): Convolutions of finite graphs and the complexity of Boolean functions. In: *Math. Logic and Appl.*, **4**, pp. 100–112. (In Russian)

[13] S. Jukna, (1986): Lower bounds on the complexity of local circuits. In: *Lect. Notes in Comput. Sci.* (Springer), vol. 233, pp. 440–448.

[14] S. Jukna, (1987a): Lower bounds on communication complexity, In: *Math. Logic and Appl.*, **5**, pp. 22–31.

[15] S. Jukna, (1987b): Information flow and width of branching programs. In: *Lect. Notes in Comput. Sci.* (Springer), vol. 287, pp. 228–230.

[16] S. Jukna, (1988a): Entropy of contact circuits and lower bounds on their complexity, *Theoret. Computer Sci.* **57**, pp. 113–129.

[17] S. Jukna, (1988b): On one entropical method of obtaining lower bounds on the complexity of Boolean functions, *Doklady Akademii Nauk SSSR*, 298:3, pp. 556-559 (In Russian)

[18] S. Jukna, (1988c): Two lower bounds for circuits over the basis $\{\wedge, \vee, \neg\}$. In: *Lect. Notes in Comput. Sci.* (Springer), vol 324, pp. 371-380.

[19] S. Jukna, (1989): The effect of null-chains on the complexity of contact circuits. In: *Lect. Notes in Comput. Sci.* (Springer), vol. 380, pp. 246-256.

[20] S. Jukna, (1990a): Monotone circuits and local computations. In: *Proc. of 31th Conf. of Lithuanian Math. Society,* (1990), pp. 100–101.

[21] S. Jukna, (1990b): Functional approximations in the theory of circuit complexity, *Dikretnaja Matematika*, **2**:2, pp. 45-59 (in Russian)

[22] S. Jukna, (1991): Optimal versus stable in boolean formulae. In: *Lect. Notes in Comput. Sci.* (Springer),vol. 529, pp. 265–274.

[23] S. Jukna, (1992): A note on read-k-times branching programs, *RAIRO Theoret. Informatics and Appl.*, vol. 29, N. 1 (1995), pp. 75-83.

[24] S. Jukna, (1994): Finite limits and lower bounds for circuit size. Tech. Rep. 94–06, Informatik, University of Trier, 1994.

[25] S. Jukna, (1995a): Computing threshold functions by depth-3 threshold circuits with smaller thresholds of their gates, *Information Processing Letters,* 56, pp. 147-150.

[26] S. Jukna, (1995b): The graph of integer multiplication is hard for read-k times networks, Tech. Rep. Nr. 95-10, University of Trier, 1995.

[27] S. Jukna, (1995c): On communication games with more than two players. Tech. Rep. 95–11, Informatik, University of Trier, 1995.

[28] S. Jukna, (1997): Finite limits and monotone computations: the lower bounds criterion, In: *Proc. of the 12th Ann. IEEE Conf. on Comput. Complexity*, pp. 302–313.

[29] S. Jukna, (1998a): Exponential Lower Bounds for Semantic Resolution, In: *Feasible Arithmetics and Length of Proofs*, P. Beame and S. Buss, Eds., DIMACS Series in Discrete Mathematics and Theoretical Comput. Sci., vol. 39 (American Math. Society, 1998), 163-172.

[30] S. Jukna, (1998b): Linear codes are hard for oblivious read-once parity branching programs. To appear in: *Information Processing Letters*.

[31] S. Jukna, (1999): Combinatorics of monotone computations, *Combinatorica*. **19** (1), pp. 65–85.

[32] S. Jukna, A. Razborov, P. Savický and I. Wegener, (1997): On P versus NP∩co-NP for decision trees and read-once branching programs, In: *Lect. Notes in Comput. Sci.* (Springer), vol. 1295, pp. 319-326. Journal version appeared in: *Computational Complexity*, **8**:4 (1999), pp. 357-370.

[33] S. Jukna and A. Razborov, (1998): Neither reading few bits twice nor reading illegally helps much, *Discrete Appl. Math.,* **85**:3, pp. 223–238.

[34] S. Jukna and S. Žák, (1998): On Branching Programs With Bounded Uncertainty, In: *Lect. Notes in Comput. Sci.* (Springer), vol. 1443, pp. 259–270. Journal version appeared in: *Theoret. Comput. Sci.* **90**:3 (2003), pp. 1851-1867.

[35] V. M. Khrapchenko, (1971): A method of determining lower bounds for the complexity of Π–schemes, *Matematicheskie Zametki*, 10:1, pp. 83–92 (in Russian). English translation in: *Mathematical Notes of the Academy of Sciences of the USSR*, 10:1, pp. 474–479.

[36] E. I. Nečiporuk, (1966): On a Boolean function, *Soviet Mathematics Doklady* **7**:4, pp. 999-1000 (In Russian)

[37] W. Paul (1977): A $2.5n$ lower bound on the combinational complexity of Boolean functions, *SIAM J. on Computing*, **6**:3, pp. 427–443.

[38] A. Razborov, (1987): Lower bounds on the monotone complexity of Boolean functions. In: *Proc. of Int. Congress of Mathematicians* (Berkeley, California, USA,1986), 1987, 1478–1487.

[39] J. A. Robinson, (1965): A machine-oriented logic based on the resolution principle, *Journal of the ACM* **12**:1, pp. 23–41.

[40] C. P. Schnorr, (1980): A 3n-lower bound on the network complexity of Boolean functions, *Theor. Comput. Sci.,* **10**, pp. 83–92.

[41] J. Simon and M. Szegedy, (1993): A new lower bound for read-only-once branching programs and its applications. In *Advances in Computational Complexity* (J. Cai, editor), AMS-DIMACS Series, vol. 13 (1993), 183–193.