Mathesis Universalis

Ján Pich

Test textu

ak ti P v kontexte K na tvrdenie T povie 1 resp 0 vezmi $f(P, K, T) = 1$ resp 0

$f(P_1, K_1, T_1) = 1/0, \ldots, f(P_i, K_j, T_k) = 1/0$ kde $T_1, \ldots, T_k$ obsahuju text S
su vahy S v korespondujucich kontextoch

najdi vahy Howl v roku 1955 najdi vahy Ariel v roku 1962
najdi vahy Dada v roku 1916

uhadni najefektivnejsi algoritmus f ktory dava tieto vahy

ak $f(1, C, B$ je velka basen$) = 1$ prehlas v kontexte C B je velka basen

ak ti P v kontexte K povie ze sa mylis vezmi $f(P, K, B$ je velka basen$) = 0$
a znovu najdi najefektivnejsie f ktore splnuje aj toto kriterium
ak $f(1, C, B$ je velka basen$) = 0$ uznaj ze si sa mylil
inak prehlas ze f je najefektivnejsim vysvetlenim a dokazom toho ze mas pravdu

ak ta (v kontexte K) poziadaju aby si napisal inovativnu basen
najdi text B tz $f(1, K, B$ je inovativna basen$) = 1$ a prehlas B

ak ti povedia F uhadni X tz $f(1, K, X$ je odpoved na F$) = 1$
a prehlas X

ak ti povedia ze pravidla ktore nasledujes su chybne a tvoja analyza to potvrdi
najdi najlepsi algoritmus ktory vyhovuje ich vyhradam a nasleduj jeho pravidla

Produkcia inovacii

I je efektivny algoritmus produkujuci inovacie
ak pre kazdy efektivny obvod S a orakulum K reprezentovane efektivnym obvodom
 definujucim otazky na ktore vie K odpovedat
 a takym ze pomocou K nejde riesit efektivne cele NP
I efektivne najde x, y tz text x splnuje (efektivne overitelne) kriteria y ale
S pouzivajuc orakulum K ziaden text splnujuci y nenajde

ak teda obvod S efektivne algoritmizuje zname sposoby produkovania textu
 a NP nejde riesit efektivnymi obvodmi
I vyprodukuje text x splnujuci kriteria y tz S nebude schopne napisat text ktory by splnil y
 ak opakovanim tohto pre rozne strategie S dostavame dvojice x, y ktore
 su predvidatelne v tom zmysle ze su popisatelne efektivnym obvodom
 reprezentujucim orakulum pomocou ktoreho nejde efektivne riesit cele
 NP a potom rozsirime S o toto orakulum
I vyprodukuje nove x, y tz S s tymto orakulom nenajde text splnujuci y atd
v tomto zmysle I produkuje vzdy invencne texty z pohladu S

da sa ukazat ze ak neexistuje malo efektivnych obvodov reprezentujucich orakula
 tz pomocou ziadneho z nich nejde efektivne riesit cele NP ale ktorych
 zjednotenim to uz ide
tak existuje efektivny obvod produkujuci inovacie
problem je tento obvod efektivne najst

Efektivne rozpoznavanie doveryhodnosti algoritmicky silnej strany

nech C tvrdi "pre kazde x plati T(x)" kde T je efektivne overitelna vlastnost x
napr "Y je najlepsia odpoved splnujuca dane efektivne overitelne kriteria"
(chceme overit ci ma C pravdu

koduj tvrdenie $\neg$T(x) ako SAT formulu s premennymi $x = x_1, \ldots, x_n$
a tu reprezentuj ako multipremenny polynom $Y(x)$ stupna $d = n^{O(1)}$
(chceme overit ci pre kazde 0/1 ohodnotenie x plati $Y(x) = 0$
(teda ci $\Sigma_{x \in 2^n} Y(x) \bmod p = 0$
(kde p je zafixovane provocislo z intervalu $(2^n, 2^{2n}]$

poziadaj C o koeficienty $< d + 1$ stupnoveho polynomu
$f(X) := \Sigma_{x_2, \ldots, x_n \in 2^{n-1}} Y(X, x_2, \ldots, x_n)$
ak C zasle polynom h tz $h(0) + h(1) \bmod p$ neni 0 prehlas "C je podozrive"
inak zvol nahodne r z intervalu $\{0, \ldots, p - 1\}$
a rekurzivne pouzi tento protokol na overenie toho ci $f(r) = h(r) \bmod p$
az kym neohodnotis vsetky $x_1, \ldots, x_n$

ak C nezavaha ani po ohodnoteni vsetkych $x_1, \ldots, x_n$
prehlas "C je doveryhodne" inak "C je nedoveryhodne"

{ak Y je najlepsia odpoved, tj $\Sigma_{x \in 2^n} Y(x) = 0$
{ existuju odpovede ktorymi nas o tom C moze presvedcit
{inak je pravdepodobnost ze odhalime C aspon $(1 - d/p)^n$
{ kedze polynom $f - h$ ma nanajvys d korenov
{ a teda pri kazdej volbe r nutime C pokracovat v klamani
{ s pravdepodobnostou aspon $1 - d/p$

problem: je mozne overit ci ma C pravdu bez
ziadania aby C riesilo viac nez NP ulohy?

Teoria zlozitosti

Je mozne pochopit a automatizovat vseobecne narocne procesy ako dokazovanie matematickych teorem ci pisanie poezie? Tieto otazky mozme dostatocne zmysluplne formulovat v jazyku teorie zlozitosti zaoberajucej sa algoritmickou narocnostou problemov.

Formalne je problem dany ako mnozina konecnych retazcov nul a jedniciek, tzv. binarne retazce. Mozu ho tvorit napriklad binarne retazce kodujuce matematicke teoremy. Riesit taky problem znamena vediet rozhodovat nejakym algoritmom ci je lubovolny dany binarny retazec v mnozine ktora problem definuje. V uvedenom priklade teda rozhodovat ci je dany retazec pravdive matematicke tvrdenie.

Zlozitost problemu meriame najcastejsie vzhladom k minimalnemu poctu krokov potrebnych na jeho riesenie nejakym algoritmom. Specialne, symbolom P oznacujeme mnozinu problemov ktore mozno riesit menej nez tzv. polynomialnym poctom krokov (nejakeho algoritmu). Z matematickeho hladiska ma mnozina P mnoho dobrych vlastnosti na to aby sa s nou pracovalo ako s aproximaciou problemov ktore ide riesit efektivne, v kratkom case. V skutocnosti ale P obsahuje tiez problemy ktore nejde riesit efektivne a naopak existuju problemy ktore su v praxi lahke a nie su v P.

Prakticky preto P nekoresponduje uplne k slovu efektivny tak ako ho pouzivame v prirodzenom jazyku. To plati aj pre mnoho dalsich konceptov a tvrdeni z teorie zlozitosti. Kedze moja motivacia pochadza z vyznamu slov daneho prave prirodzenym jazykom prezentovane basne su formulovane prevazne v nom.

Druhou vyznamnou mnozinou problemov je NP. Tvoria ju problemy ktorych riesenie je mozne efektivne overit. Napriklad dokazovanie matematickych teorem mozme formulovat ako NP problem pretoze otazka ci je dane tvrdenie (v praxi dokazatelna) teorema ma efektivne overitelne riesenie, ktorym je (kratky) dokaz. Nadnesene sa da povedat ze NP obsahuje vsetky problemy. Ak totiz mame problem ktoreho riesenie nejde efektivne overit, da sa pochybovat o jeho zmysluplnosti.

Fundamentalnym otvorenym problemom teorie zlozitosti je otazka ci plati P=NP, teda zjednodusene otazka ci je mozne efektivne najst riesenie problemu ak nejake lahko overitelne riesenie existuje. Dnes nedokazeme popriet existenciu efektivnych algoritmov ktore by dokazali v okamihu riesit NP problemy a specialne aj matematicke teoremy.

Ako zlozite je teda nachadzanie odpovedi na prakticky vsetky otazky, je mozne pochopit a automatizovat tak kreativny proces ako je dokazovanie matematickych teorem alebo tvorba poezie?

Basen Test textu ilustruje algoritmus na pisanie poezie, ktory je "v tzv. polynomialnej hierarchii". Ak P=NP (korektnejsie, ak existuje efektivny algoritmus pre NP problemy), tento algoritmus mozno simulovat efektivne.

Riesit vsetky NP problemy efektivne mozno nejde, ale aj dokaz toho ze P nie je NP moze mat podobne dosledky. Dostatocne konstruktivna separacia P a NP by totiz davala efektivny algoritmus dosvedcujuci chyby potencionalnych efektivnych algoritmov pre NP problemy, vid Definicia 1 nizsie. Dosvedcit chybu algoritmu tu znamena najst riesenie nejakej otazky, ktoru dany algoritmus nevie zodpovedat spravne. Z pohladu chybujuceho algoritmu je take riesenie akoby inovativnym textom (vymykajucim sa predoslym sposobom produkovania rieseni). V basni Produkcia inovacii je prezentovany algoritmus generujuci inovacie tak aby fungoval navyse proti istym orakulam vynucujucim dostatocnu roznorodost inovacii, vid Definicia 2.

Aj takyto konstruktivny dokaz toho ze P nie je NP moze byt tazke najst. Preto ma zmysel klast si potencialne dosiahnutelnejsie ciele. Je napriklad mozne efektivne overit presvedcenie, ze dalsi bit basne ma byt 0 ci 1? Schopnost rychlo overit jeho doveryhodnost a zachovat sa tak najlepsie v ramci moznosti by bola podobne uzitocna ako samotne efektivne nachadzanie dalsieho bitu poezie. Basen Efektivne rozpoznavanie doveryhodnosti.. popisuje taky test, ktory je aplikaciou znameho vysledku teorie zlozitosti, tzv. IP protokolu pre coNP problemy. Jeho nevyhodou ale je, ze vyzaduje aby testovana strana riesila prilis narocne problemy oznacovane ako #P.

Hierarchiu problemov teorie zlozitosti naznacenu v predchadzajucom texte by slo rozvijat dalej. Uz jej pociatocne otazky pritom ostavaju nezodpovedane.

Definicia 1 *Nech k je konstanta. F je efektivny algoritmus dosvedcujuci chyby Booleovych obvodov velkosti n^k pokusajucich sa riesit NP problemy, ak pre kazde n a kazdy obvod C s n vstupmi a velkostou n^k, F najde v polynomialnom case vyrokovu formulu x velkosti n a jej splnujuce ohodnotenie y pricom x nie je splnena ohodnotenim $C(x)$.*

Ak by sme definovali inovativny text ako lubovolny text T, pre ktory existuje nejake efektivne overitelne kriterium, ktore T splnuje, a ktore nejde splnit predoslymi sposobmi "tvorenia" poezie (tieto sposoby by boli dane najmensim obvodom, ktory dokaze produkovat texty splnujuce kriteria C pre kazde efektivne overitelne C splnene nejakym textom predchadzajucim T), bol by aj nahodny text s velkou pravdepodobnostou inovativny (predpokladajuc existenciu jednosmernych funkcii):

kriterium dosvedcujuce invencnost nahodneho textu x by bolo $f(y) = f(x)$,
kde f je jednosmerna funkcia a y su volne premenne
(ktorych hodnoty treba pre splnenie kriteria $f(y) = f(x)$ najst),
konkretnejsie, napr. pre nahodne dost velke prvocisla p, q by bol
text $pq = n$ invencny pretoze by slo o faktorizaciu cisla n,
co je problem ktory nevieme efektivne riesit.

Definicia 2 (Algoritmus z basne Produkcia Inovacii formalne) *Nech k, l su konstanty. F je efektivny algoritmus produkujuci inovacie voci Booleovym obvodom velkosti n^k a orakulam velkosti n^l, ak F vzdy zastavi v polynomialnom case a pre kazde n, kazdy obvod C s n vstupmi a velkostou n^k, a kazdy obvod D s n vstupmi a velkostou n^l taky, ze*

SAT neni v P^A pre orakulum A schopne nachadzat splnujuce ohodnotenia (ak existuju) formul x splnujucich $D(x) = 1$,

plati, ze $F(C, D) = \langle x, y \rangle$, kde x je vyrokova formula velkosti n splnena ohodnotenim y ale nesplnena ohodnotenim ktore na vstupe x vyprodukuje obvod C pouzivajuc orakulum A.

Mathesis universalis

automatizovat poznavanie a tvorivy proces
vziat tvrdenie (prepisat ho formalne) rozbehnut mechanicky kalkul
a rozhodnut jeho pravdivost
takto postupne rozhodovat co ma byt dalsi bit basne a celu ju najst

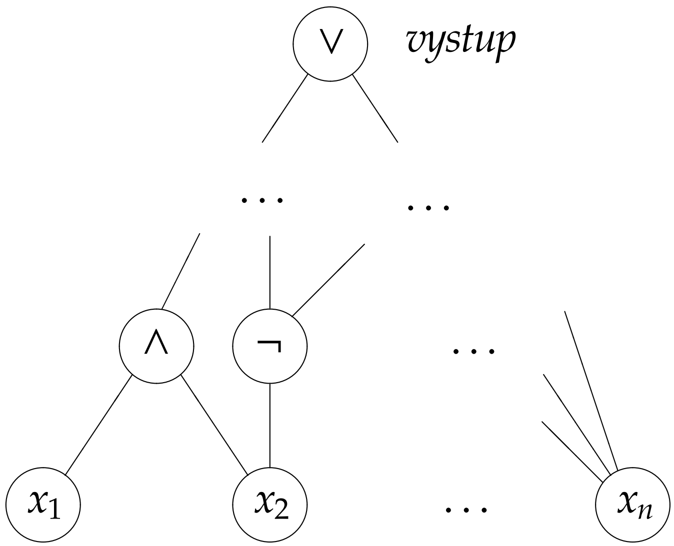
okrem neuplnosti dostatocne silnych konzistentnych fragmentov matematiky je ale problematicka uz i formalizacia beznych tvrdeni, nie je jasne ako definovat koncepty ako inovativnost boh apod

obideme tieto problemy tym ze ostaneme v prirodzenom jazyku zakodovanom do binarnych postupnosti a budeme s nim operovat v kalkule vyrokovej logiky
ktora je uplna

popiseme tento proces preciznejsie

---

zafixujme akekolvek standardne kodovanie prirodzeneho jazyka do
binarnych postupnosti
napr "a" je 0001, "b" 0010, " " 0000 atd, text "ba a" je potom 0010000100000001
vlastnost (resp tvrdenie o) binarnej postupnosti x mozme vyjadrit ako tvrdenie $D(x)$ pomocou vhodneho obvodu D pozostavajuceho z logickych spojok AND ($\wedge$), OR ($\vee$) a NOT ($\neg$):



napr $(x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2)$ tvrdi ze v postupnosti x_1, x_2 je prave jedna 1

podobne mozme vyjadrit vlastnosti premennych C a y ako:
"ak C koduje obvod tak ten na vstupe y dava 1", v skratke $C(y) = 1$
(C je tu skratka pre $c_1, \ldots, c_m$, podobne $y = y_1, \ldots, y_n$)

specialne nas budu zaujimat tvrdenia typu:
"ak (uzivatel) T tvrdi/odmieta x potom $C(T, x) = 1/0$", kde C je znova
kodovane ako neznamy obvod
napr "ak y koduje axiom ci znamu teoremu ZFC[1] tak $C(y) = 1$ a $C(\neg y) = 0$"
"ak T tvrdi ze *Invocation of laughter* je/nie je invencna basen
tak $C(T, \textit{Invocation of laughter}) = 1/0$"
po dost velkom mnozstve takychto axiom sa C "nauci" ako reagovat, najmensie C splnujuce vsetky axiomy dane predoslymi skusenostami mozme interpretovat ako ich najefektivnejsie vysvetlenie

vysvetlenie C je neznamy objekt a je problem ho najst
v skutocnosti ho ale najst nepotrebujeme, hoci ide o neznamy objekt
mozme s nim kalkulovat
a staci ak pouzitim jeho vlastnosti odvodime tvrdenia ktore nas zaujimaju
napr "ak obvod C splna $C(b) = 1/0$ pre basne b podla konkretneho navrhu
poetickeho kanonu (a kazde mensie C nesplna niektoru z tychto axiom)
tak $C(\textit{Ariel}) = 1?$"
"ak C splna axiomy a zname teoremy ZFC (a mensie C v tom zlyhavaju)
tak $C(\textit{Hypoteza Kontinua}) = 1?$"

---

korektne usudit tvrdenie $A(x)$ z tvrdeni $B_1(x), \ldots, B_i(x)$ mozme ak
kazde x splnujuce $B_1(x)$ az $B_i(x)$ splnuje tiez $A(x)$
prikladom korektneho odvodzovacieho pravidla je modus ponens:
ak mame $B(x)$ a $B(x) \to A(x)$ mozme odvodit $A(x)$

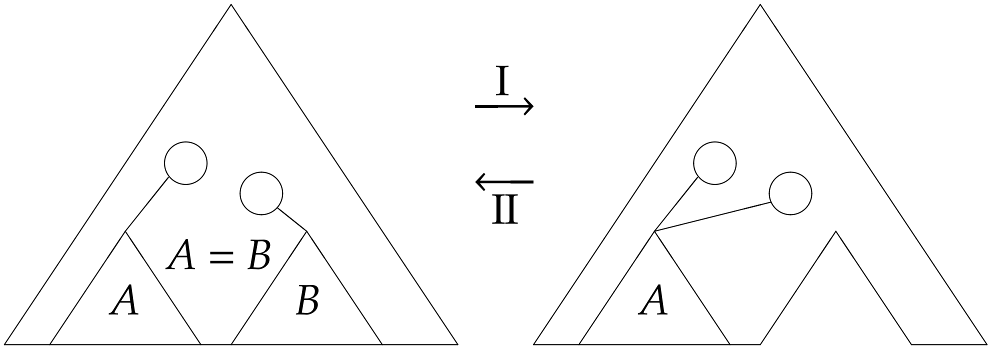
konecna mnozina takych pravidiel tvori dokazovy system ak
pomocou nich mozme odvodit kazde pravdive tvrdenie (platne pre kazde x)
napr modus ponens, axiomy (ktore mozme vidiet ako pravidla):

$$A \to (B \to A)$$
$$(A \to (B \to C)) \to ((A \to B) \to (A \to C))$$
$$(\neg A \to \neg B) \to (B \to A)$$

a pravidla I, II ilustrovane nizsie tvoria taky system

[1] ZFC je teoria formalizujuca prakticky vsetku beznu matematiku

I. umoznuje nahradit dva identicke podobvody jednym
II. umoznuje rozdvojit podobvod pouzity na dvoch roznych miestach



(pozn.: $\wedge$ a $\vee$ sa daju definovat pomocou $\to$ a $\neg$)
tj vsetko co mozme vydedukovat mozme vydedukovat uz pomocou modus ponens a par zmienenych pravidiel, specialne mozme nimi simulovat vsetky ostatne odvodzovacie pravidla

ak je ale odvodenie tvrdenia exponencialne dlhe je prakticky nerealizovatelne
ako rychlo mozme dedukovat pravdive tvrdenia?
mozme pravidlami uvedenymi vyssie odvodit kazde pravdive tvrdenie ktore pozostava z n symbolov dokazom ktory ma symbolov nanajvys $10n^2$?

Selected open problems in proof complexity

1. ma EF polynomialne kratke dokazy pravdivych tvrdeni?[1]
   ak nie, ziaden polynomialny algoritmus neriesi NP dokazatelne v EF[2]
   ktore pravdive tvrdenia maju kratke dokazy?

2. je EF optimalny dokazovy system? (existuje optimalny dokazovy system?)[3]
   ekvivalentne: maju efektivne generovatelne resp rozpoznatelne tautologie kratke EF dokazy?[4]
   (kratke dokazy v nejakom inom dokazovom systeme?)
   - 2.1. dokazuje ZFC tautologie efektivnejsie nez EF?[5]

3. na ktorych tvrdeniach sa da EF automatizovat?
   tj pre ktore typy tvrdeni existuje algoritmus nachadzajuci EF dokazy efektivne vzhladom k ich dlzke?
   - 3.1. pre vsetky efektivne generovatelne tautologie existuje algoritmus nachadzajuci EF dokazy efektivne vzhladom k ich dlzke?
   - 3.2. je EF p-optimalny?
     tj pre vsetky efektivne generovatelne tautologie existuje algoritmus nachadzajuci EF dokazy efektivne?

---

[1] EF je zauzivana obdoba systemu definovaneho v texte Mathesis universalis (modus ponens, pravidla I, II a prislusne axiomy), tu mozme na jeho mieste brat prave system z Mathesis universalis. EF ma polynomialne kratke dokazy tautologii ak existuje k tz kazda tautologia pozostavajuca z n symbolov ma EF dokaz s nanajvys kn^k symbolmi.

[2] Pre ziaden polynomialny algoritmus f EF nema polynomialne kratke dokazy tvrdeni $SAT(x, y) \rightarrow SAT(x, f(x))$. $SAT(x, y)$ znamena ze formula x je splnena ohodnotenim premennych y, tj tvrdenie x o premennych z plati ak $z = y$.

[3] Vseobecne, dokazovy system je akykolvek efektivny algoritmus A ktory pre kazde x splnuje ekvivalenciu: x je tautologia prave vtedy ak existuje y tz $A(x, y) = 1$. Dokazovy system A je optimalny ak pre kazdy system B existuje k tz kazda tautologia s dokazom dlzky s v systeme B ma dokaz dlzky ks^k v systeme A.

[4] Postupnost tautologii $\phi_1, \phi_2, \ldots$ je efektivne generovatelna ak existuje efektivny algoritmus ktory pre kazdy retazec dlzky n vyprodukuje ϕ_n.

[5] Neexistuje k tz kazda tautologia so ZFC dokazom dlzky s ma EF dokaz dlzky ks^k?

4. maju fundamentalne otazky teorie zlozitosti kratke EF riesenia?

napr $\mathsf{lb}(\mathrm{SAT}, n^k)$: SAT nejde riesit obvodmi velkosti n^k

$\mathsf{sprng}(g, n^k)$: g je pseudonahodny generator pre obvody velkosti n^k

$\mathsf{oneway}(f, n^k)$: funkciu f je tazke invertovat obvodmi velkosti n^k

$\mathsf{eflb}(T, n^k)$: tvrdenie T velkosti n nema EF dokaz dlzky n^k

. . . [6]

5. daju sa efektivne generovat tazke tautologie?

formalne: nech n je dost velke, da sa pre kazdy efektivny obvod $C(x, y)$
s n vstupmi x a n^k vstupmi y tz $C(x, y) = 1$ len ak je x tautologia
najst efektivne tautologia x tz $C(x, y) = 0$ pre kazde y?

6. ak EF dokazuje efektivne $A \vee B$, dokazuje efektivne A alebo B?

ak ano a existuju tzv super-bity, $\mathsf{lb}(\mathrm{SAT}, n^k)$ nema kratke EF dokazy
ma EF ine podobne konstruktivne vlastnosti?

7. existuje generator g tazky pre vsetky dokazove systemy?

tj existuje funkcia $g : \{0, 1\}^n \mapsto \{0, 1\}^{n+1}$ zobrazujuca binarne retazce dlzky n efektivne na binarne retazce dlzky $n + 1$ tz pre kazdy retazec b dlzky $n + 1$ tvrdenie $\forall x_1, \ldots, x_n,\ b \neq g(x_1, \ldots, x_n)$ nema kratky dokaz v ziadnom dokazovom systeme?

[6] Uvedene tvrdenia sa daju efektivne vyjadrit ako tautologie za standardnych predpokladov z teorie zlozitosti, u pravdepodobnostnych tvrdeni sa pouzije aproximacia.

Content