MATHESIS UNIVERSALIS

JÁN PICH

Mathesis Universalis

Ján Pich

Literis

Literis
ISBN 978-80-971502-2-8

Test textu

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

$\mathsf{f}(\mathrm{P}_1, \mathrm{K}_1, \mathrm{T}_1) = 1/0, \ldots, \mathsf{f}(\mathrm{P}_\mathsf{i}, \mathrm{K}_\mathsf{j}, \mathrm{T}_\mathsf{k}) = 1/0$, kde $\mathrm{T}_1, \ldots, \mathrm{T}_\mathsf{k}$ obsahujú text S,
sú váhy S v korešpondujúcich kontextoch

nájdi váhy Howl v roku 1955, nájdi váhy Ariel v roku 1962
nájdi váhy Dada v roku 1916

uhádni najefektívnejší algoritmus f, ktorý dáva tieto váhy

ak f(1,C,B je veľká báseň) = 1, vyhlás, v kontexte C, B je veľká báseň

ak ti P v kontexte K povie, že sa mýliš, vezmi f(P,K,B je veľká báseň) = 0
a znovu nájdi najefektívnejšie f, ktoré spĺňa aj toto kritérium
ak f(1,C,B je veľká báseň) = 0, uznaj, že si sa mýlil
inak vyhlás, že f je najefektívnejším vysvetlením a dôkazom toho, že máš pravdu

ak ťa (v kontexte K) požiadajú, aby si napísal inovatívnu báseň,
nájdi taký text B, že f(1,K,B je inovatívna báseň) = 1 a vyhlás B

ak ti povedia F, uhádni také X, že f(1,K,X je odpoveď na F) = 1
a vyhlás X

ak ti povedia, že pravidlá, ktoré nasleduješ, sú chybné, a tvoja analýza to potvrdí,
nájdi najlepší algoritmus, ktorý vyhovuje ich výhradám, a nasleduj jeho pravidlá

Produkcia inovácií

I je efektívny algoritmus produkujúci inovácie,
ak pre každý efektívny obvod S a orákulum K reprezentované efektívnym obvodom
    definujúcim otázky, na ktoré vie K odpovedať,
    a takým, že pomocou K sa nedá riešiť efektívne celé NP,
I efektívne nájde také x, y, že text x spĺňa (efektívne overiteľné) kritériá y, ale S používajúc orákulum K žiaden text spĺňajúci y nenájde.

Ak teda obvod S efektívne algoritmizuje známe spôsoby produkovania textu
    a NP sa nedá riešiť efektívnymi obvodmi,
I vyprodukuje text x spĺňajúci také kritériá y, že S nebude schopné napísať text, ktorý by splnil y.
    Ak opakovaním tohto pre rôzne stratégie S dostávame dvojice x, y, ktoré sú predvídateľné v tom zmysle, že sú popísateľné efektívnym obvodom reprezentujúcim orákulum, pomocou ktorého sa nedá efektívne riešit celé NP, a potom rozšírime S o toto orákulum,
I vyprodukuje nové x, y, pre ktoré S s týmto orákulom nenájde text spĺňajúci y atď.
V tomto zmysle I produkuje vždy invenčné texty z pohľadu S.

Dá sa ukázať, že ak neexistuje málo efektívnych obvodov reprezentujúcich orákula
    s vlastnosťou, že pomocou žiadneho z nich nie je možné efektívne riešiť celé NP, ale ktorých zjednotením to už možné je,
tak existuje efektívny obvod produkujúci inovácie.
Problém je tento obvod efektívne nájsť.

Efektívne rozpoznávanie dôveryhodnosti algoritmicky silnej strany

nech C tvrdí: „pre každé x platí $T(x)$“, kde T je efektívne overiteľná vlastnosť x,
napr. „Y je najlepšia odpoveď spĺňajúca dané efektívne overiteľné kritériá“
(chceme overiť, či má C pravdu

kóduj tvrdenie $\neg T(x)$ ako SAT formulu s premennými $x = x_1, \ldots, x_n$
a tú reprezentuj ako multipremenný polynóm $Y(x)$ stupňa $d = n^{O(1)}$
(chceme overiť, či pre každé 0/1 ohodnotenie x platí $Y(x) = 0$
(teda či $\Sigma_{x \in 2^n} Y(x)\ mod\ p = 0$
(kde p je zafixované provočíslo z intervalu $(2^n, 2^{2n}]$

požiadaj C o koeficienty $< d + 1$ stupňového polynómu
$f(X) := \Sigma_{x_2,\ldots,x_n \in 2^{n-1}} Y(X, x_2, \ldots, x_n)$
ak C zašle taký polynóm h, že $h(0)+h(1)\ mod\ p$ nie je 0, vyhlás: „C je podozrivé“
inak zvoľ náhodné r z intervalu $\{0, \ldots, p-1\}$
a rekurzívne použi tento protokol na overenie toho, či $f(r) = h(r)\ mod\ p$
až kým neohodnotíš všetky $x_1, \ldots, x_n$

ak C nezaváha ani po ohodnotení všetkých $x_1, \ldots, x_n$,
vyhlás: „C je dôveryhodné“, inak „C je nedôveryhodné“

{ak Y je najlepšia odpoveď, t. j. $\Sigma_{x \in 2^n} Y(x) = 0$
{ existujú odpovede, ktorými nás o tom C môže presvedčiť
{inak je pravdepodobnosť, že odhalíme C, aspoň $(1 - d/p)^n$
{ keďže polynóm $f - h$ má nanajvýš d koreňov
{ a teda pri každej voľbe r nútime C pokračovať v klamaní
{ s pravdepodobnosťou aspoň $1 - d/p$

problém: je možné overiť, či má C pravdu, bez
žiadania, aby C riešilo viac než NP úlohy?

Teória zložitosti

Je možné pochopiť a automatizovať všeobecne náročné procesy ako dokazovanie matematických teorém či písanie poézie? Tieto otázky môžeme dostatočne zmysluplne formulovať v jazyku teórie zložitosti zaoberajúcej sa algoritmickou náročnosťou problémov.

Formálne je problém daný ako množina konečných reťazcov núl a jednotiek, tzv. binárne reťazce. Môžu ho tvoriť napríklad binárne reťazce kódujúce matematické teorémy. Riešiť taký problém znamená vedieť rozhodovať nejakým algoritmom, či je ľubovoľný daný binárny reťazec v množine, ktorá problém definuje. V uvedenom príklade teda rozhodovať, či je daný reťazec pravdivé matematické tvrdenie.

Zložitosť problému meriame najčastejšie vzhľadom na minimálny počet krokov potrebných na jeho riešenie nejakým algoritmom. Symbolom P špeciálne označujeme množinu problémov, ktoré možno riešiť menej než tzv. polynomiálnym počtom krokov (nejakého algoritmu). Z matematického hľadiska má množina P mnoho dobrých vlastností na to, aby sa s ňou pracovalo ako s aproximáciou problémov, ktoré je možné riešiť efektívne, v krátkom čase. V skutočnosti ale P obsahuje tiež problémy, ktoré nie je možné riešiť efektívne a, naopak, existujú problémy, ktoré sú v praxi ľahké a nie sú v P.

Prakticky preto P nekorešponduje úplne so slovom efektívny tak, ako ho používame v prirodzenom jazyku. To platí aj pre mnoho ďalších konceptov a tvrdení z teórie zložitosti. Keďže moja motivácia pochádza z významu slov daného práve prirodzeným jazykom, prezentované básne sú formulované prevažne v ňom.

Druhou významnou množinou problémov je NP. Tvoria ju problémy, ktorých riešenie je možné efektívne overiť. Napríklad dokazovanie matematických teorém môžeme formulovať ako NP problém, pretože otázka, či je dané tvrdenie (v praxi dokázateľná) teoréma, má efektívne overiteľné riešenie, ktorým je (krátky) dôkaz. Nadnesene sa dá povedať, že NP obsahuje všetky problémy. Ak totiž máme problém, ktorého riešenie nie je možné efektívne overiť, dá sa pochybovať o jeho zmysluplnosti.

Fundamentálnym otvoreným problémom teórie zložitosti je otázka, či platí P=NP, teda zjednodušene otázka, či je možné efektívne nájsť riešenie problému, ak nejaké ľahko overiteľné riešenie existuje. Dnes nedokážeme poprieť existenciu efektívnych algoritmov, ktoré by dokázali v okamihu riešiť matematické teorémy a ostatné bežné NP problémy.

Aké zložité je teda nachádzanie odpovedí na prakticky všetky otázky, je možné pochopiť a automatizovať taký kreatívny proces, ako je dokazovanie matematických teorém alebo tvorba poézie?

Báseň *Test textu* ilustruje algoritmus na písanie poézie, ktorý možno simulovať efektívne, ak P=NP (korektnejšie, ak existuje efektívny algoritmus pre problémy s efektívne verifikovateľnou odpoveďou).

Riešiť NP problémy v polynomiálnom počte krokov sa možno nedá, ale aj dôkaz toho, že P nie je NP, môže mať podobné dôsledky. Dostatočne konštruktívna separácia P a NP by totiž dávala efektívny algoritmus dosvedčujúci chyby potenciálnych efektívnych algoritmov pre NP problémy, pozri Definíciu 1 nižšie. Dosvedčiť chybu algoritmu tu znamená nájsť riešenie nejakej otázky, ktorú daný algoritmus nevie zodpovedať správne. Z pohľadu chybujúceho algoritmu je také riešenie akoby inovatívnym textom (vymykajúcim sa predošlým spôsobom produkovania riešení). V básni *Produkcia inovácií* je prezentovaný algoritmus generujúci inovácie tak, aby fungoval navyše proti istým orákulám vynucujúcim dostatočnú rôznorodosť inovácií, pozri Definíciu 2.

Aj takýto konštruktívny dôkaz toho, že P nie je NP, môže byť ťažké nájsť. Preto má zmysel klásť si potenciálne dosiahnuteľnejšie ciele. Je napríklad možné efektívne overiť presvedčenie, že ďalší bit básne má byť 0 či 1? Schopnosť rýchlo overiť jeho dôveryhodnosť a zachovať sa tak najlepšie v rámci možností by bola podobne užitočná ako samotné efektívne nachádzanie ďalšieho bitu poézie. Báseň *Efektívne rozpoznávanie dôveryhodnosti* popisuje taký test, ktorý je aplikáciou známeho výsledku teórie zložitosti, tzv. IP protokolu pre coNP problémy. Jeho nevýhodou ale je, že vyžaduje, aby testovaná strana riešila príliš náročné problémy označované ako #P.

Hierarchiu problémov teórie zložitosti naznačenú v predchadzájucom texte by šlo rozvíjať ďalej. Už jej počiatočné otázky pritom ostávajú nezodpovedané.

Definícia 1 *Nech k je konštanta. F je efektívny algoritmus dosvedčujúci chyby Booleových obvodov veľkosti n^k pokúšajúcich sa riešiť NP problémy, ak pre každé n a každý obvod C s n vstupmi a veľkosťou n^k F nájde v polynomiálnom čase výrokovú formulu x veľkosti n a jej spĺňajúce ohodnotenie y, pričom x nie je splnená ohodnotením $C(x)$.*

Ak by sme definovali inovatívny text ako ľubovoľný text T, pre ktorý existuje nejaké efektívne overiteľné kritérium, ktoré T spĺňa a ktoré sa nedá splniť predošlými spôsobmi „tvorenia“ poézie (tieto spôsoby by boli dané najmenším obvodom, ktorý dokáže produkovať texty spĺňajúce kritéria C pre každé efektívne overiteľné C splnené nejakým textom predchádzajúcim T), bol by aj náhodný text s veľkou pravdepodobnosťou inovatívny (predpokladajúc existenciu jednosmerných funkcií):

kritérium dosvedčujúce invenčnosť náhodneho textu x by bolo $f(y) = f(x)$,
kde f je jednosmerná funkcia a y sú voľné premenné
(ktorých hodnoty treba na splnenie kritéria $f(y) = f(x)$ nájsť),
konkrétnejšie, napr. pre náhodné dosť veľké prvočísla p, q by bol
text $pq = n$ invenčný, pretože by šlo o faktorizáciu čísla n,
čo je problém, ktorý nevieme efektívne riešiť.

Definícia 2 (Algoritmus z básne *Produkcia inovácií* formálne) *Nech k, l sú konštanty. F je efektívny algoritmus produkujúci inovácie voči Booleovým obvodom veľkosti n^k a orákulám veľkosti n^l, ak F vždy zastaví v polynomiálnom čase a pre každé n, každý obvod C s n vstupmi a veľkosťou n^k a každý obvod D s n vstupmi, veľkosťou n^l a vlastnosťou, že*

SAT nie je v P^A pre orákulum A schopné nachádzať spĺňajúce ohodnotenia (ak existujú) formúl x spĺňajúcich $D(x) = 1$,

platí, že $F(C, D) = \langle x, y \rangle$, kde x je výroková formula veľkosti n splnená ohodnotením y, ale nesplnená ohodnotením, ktoré na vstupe x vyprodukuje obvod C, používajúc orákulum A.

Mathesis universalis

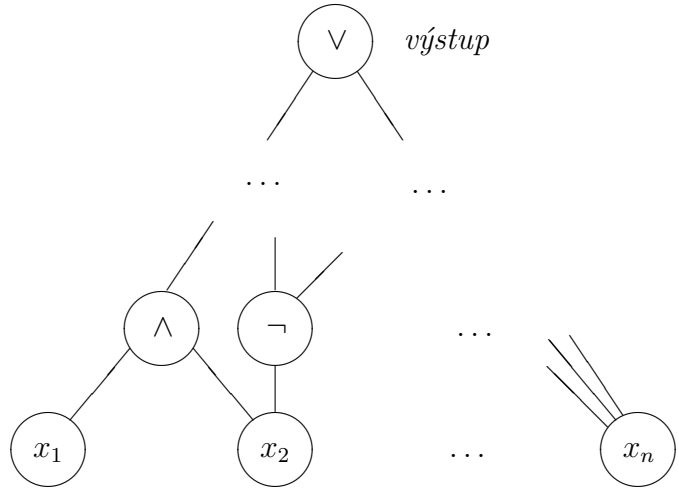
Automatizovať poznávanie a tvorivý proces.
Vziať tvrdenie (prepísať ho formálne), rozbehnúť mechanický kalkul
a rozhodnúť jeho pravdivosť.
Takto postupne rozhodovať, čo má byť ďalší bit básne a celú ju nájsť.

Okrem neúplnosti dostatočne silných, konzistentných fragmentov matematiky je ale problematická už i formalizácia bežných tvrdení. Nie je jasné ako definovať koncepty ako inovatívnosť, boh a pod.

Obídeme tieto problémy tým, že ostaneme v prirodzenom jazyku zakódovanom do binárnych postupností a budeme s ním operovať v kalkule výrokovej logiky,
ktorá je úplná.

Popíšeme tento proces precíznejšie.

Zafixujme akékoľvek štandardné kódovanie prirodzeného jazyka do
binárnych postupností,
napr. „a“ je 0001, „b“ 0010, „ “ 0000 atď., text „ba a“ je potom 0010000100000001.
Vlastnosť (resp. tvrdenie o) binárnej postupnosti x môžeme vyjadriť ako tvrdenie $D(x)$ pomocou vhodného obvodu D, pozostávajúceho z logických spojok AND ($\wedge$), OR ($\vee$) a NOT ($\neg$):



napr. $(x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2)$ tvrdí, že v postupnosti x_1, x_2 je práve jedna 1.

Podobne môžeme vyjadriť vlastnosti premenných C a y ako:
„ak C kóduje obvod, tak ten na vstupe y dáva 1“, v skratke $C(y) = 1$
(C je tu skratka pre $c_1, \ldots, c_m$, podobne $y = y_1, \ldots, y_n$).

Špeciálne nás budú zaujímať tvrdenia typu:
„ak (užívateľ) T tvrdí/odmieta x, potom $C(T, x) = 1/0$“, kde C je znova kódované ako neznámy obvod.
Napr. „Ak y kóduje axióm či známu teorému ZFC[1], tak $C(y) = 1$ a $C(\neg y) = 0$.“
„Ak T tvrdí, že *Invocation of laughter* je/nie je invenčná báseň, tak $C(T, \mathit{Invocation\ of\ laughter}) = 1/0$.“
Po dosť veľkom množstve takýchto axióm sa C „naučí“ reagovať, najmenšie C spĺňajúce všetky axiómy dané predošlými skúsenosťami môžeme interpretovať ako ich najefektívnejšie vysvetlenie.

Vysvetlenie C je neznámy objekt a je problém ho nájsť.
V skutočnosti ho ale nájsť nepotrebujeme. Hoci ide o neznámy objekt, môžeme s ním kalkulovať,
a stačí, ak použitím jeho vlastností odvodíme tvrdenia, ktoré nás zaujímajú.
Napr. „Ak obvod C spĺňa $C(b) = 1/0$ pre básne b podľa konkretného návrhu poetického kánonu (a každé menšie C nespĺňa niektorú z týchto axióm), tak $C(\mathit{Ariel}) = 1$?“
„Ak C spĺňa axiómy a známe teorémy ZFC (a menšie C v tom zlyhávajú), tak $C(\mathit{Continuum\ hypothesis}) = 1$?“

Korektne odvodiť tvrdenie $A(x)$ z tvrdení $B_1(x), \ldots, B_i(x)$ môžeme, ak každé x spĺňajúce $B_1(x)$ až $B_i(x)$ spĺňa tiež $A(x)$.
Príkladom korektného odvodzovacieho pravidla je modus ponens:
ak máme $B(x)$ a $B(x) \to A(x)$, môžeme odvodiť $A(x)$.

Konečná množina takých pravidiel tvorí dôkazový systém, ak pomocou nich môžeme odvodiť každé pravdivé tvrdenie (platné pre každé x).
Napr. modus ponens, axiómy (ktoré môžeme vidieť ako pravidlá):
$A \to (B \to A)$

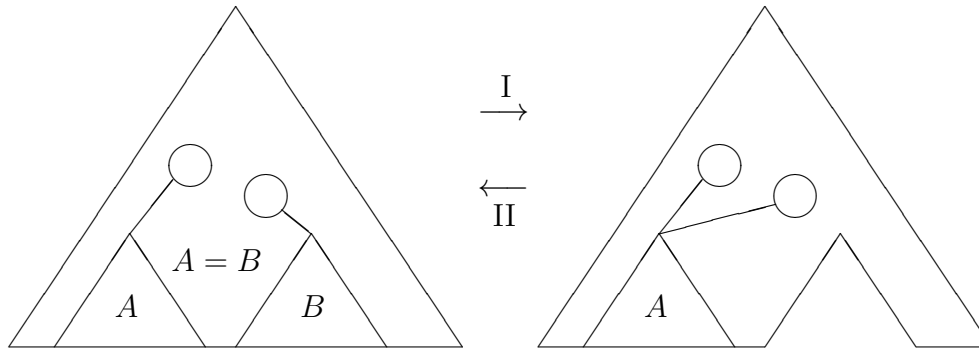
[1] ZFC je teória formalizujúca prakticky všetku bežnú matematiku.

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

a pravidlá I, II ilustrované nižšie tvoria taký systém.

I. Umožňuje nahradiť dva identické podobvody jedným.

II. Umožňuje rozdvojiť podobvod použitý na dvoch rôznych miestach.



(Pozn.: $\wedge$ a $\vee$ sa dajú definovať pomocou $\to$ a $\neg$.)

T. j. všetko, čo môžeme vydedukovať, môžeme vydedukovať už pomocou modus ponens a pár zmienených pravidiel. Konkrétne nimi môžeme simulovať všetky ostatné odvodzovacie pravidlá.

Ak je ale odvodenie tvrdenia exponenciálne dlhé, je prakticky nerealizovateľné. Ako rýchlo môžeme dedukovať pravdivé tvrdenia?

Môžeme pravidlami uvedenými vyššie odvodiť každé pravdivé tvrdenie, ktoré pozostáva z n symbolov dôkazom, ktorý má symbolov nanajvýš $10n^2$?

Vybrané otvorené problémy dôkazovej zložitosti

1. Má EF polynomiálne krátke dôkazy pravdivých tvrdení?[1]
   Ak nie, žiaden polynomiálny algoritmus nerieši NP dokázateľne v EF.[2]
   Ktoré pravdivé tvrdenia majú krátke dôkazy?

2. Je EF optimálny dôkazový systém? (Existuje optimálny dôkazový systém?)[3]
   Ekvivalentne: Majú efektívne generovateľné, resp. rozpoznateľné tautológie krátke EF dôkazy?[4]
   (Krátke dôkazy v nejakom inom dôkazovom systéme?)
   2.1. Dokazuje ZFC tautológie efektívnejšie než EF?[5]

3. Na ktorých tvrdeniach sa dá EF automatizovať?
   T. j. pre ktoré typy tvrdení existuje algoritmus nachádzajúci EF dôkazy efektívne vzhľadom na ich dĺžku?
   3.1. Existuje pre každú postupnosť efektívne generovateľných tautológií algoritmus nachádzajúci EF dôkazy efektívne vzhľadom na ich dĺžku?
   3.2. Je EF p-optimálny?
   T. j. existuje pre každú postupnosť efektívne generovateľných tautológií algoritmus nachádzajúci ich EF dôkazy efektívne?

---

[1] EF je zaužívaná obdoba systému definovaného v texte *Mathesis universalis* (modus ponens, pravidlá I, II a príslušné axiómy). Tu môžeme namiesto EF použiť práve systém z *Mathesis universalis*. EF má polynomiálne krátke dôkazy tautológií, ak existuje také k, že každá tautológia pozostávajúca z n symbolov má EF dôkaz s nanajvýš kn^k symbolmi.

[2] Pre žiaden polynomiálny algoritmus f nemá EF polynomiálne krátke dôkazy tvrdení $SAT(x, y) \rightarrow SAT(x, f(x))$. $SAT(x, y)$ znamená, že formula x je splnená ohodnotením premenných y, t. j. tvrdenie x o premenných z platí, ak $z = y$.

[3] Všeobecne, dôkazový systém je akýkoľvek efektívny algoritmus A, ktorý pre každé x spĺňa ekvivalenciu: x je tautológia práve vtedy, ak existuje také y, že $A(x, y) = 1$. Dôkazový systém A je optimálny, ak pre každý systém B existuje také k, že každá tautológia s dôkazom dĺžky s v systéme B má dôkaz dĺžky ks^k v systéme A.

[4] Postupnosť tautológií $\phi_1, \phi_2, \ldots$ je efektívne generovateľná, ak existuje efektívny algoritmus, ktorý pre každý reťazec dĺžky n vyprodukuje ϕ_n.

[5] Neexistuje také k, že každá tautológia so ZFC dôkazom dĺžky s má EF dôkaz dĺžky ks^k?

4. Majú fundamentálne otázky teórie zložitosti krátke EF riešenia?
   Napr. $\mathsf{lb}(\mathrm{SAT}, n^k)$: SAT nie je možné riešiť obvodmi veľkosti n^k
   $\mathsf{sprng}(g, n^k)$: g je pseudonáhodný generátor pre obvody veľkosti n^k
   $\mathsf{oneway}(f, n^k)$: funkciu f je ťažké invertovať obvodmi veľkosti n^k
   $\mathsf{eflb}(T, n^k)$: tvrdenie T veľkosti n nemá EF dôkaz dĺžky n^k
   ... [6]

5. Dajú sa efektívne generovať ťažké tautológie?
   Formálne: Nech n je dosť veľké. Dá sa pre každý efektívny obvod $C(x, y)$ s n vstupmi x a n^k vstupmi y, ktorý spĺňa $C(x, y) = 1$, len ak je x tautológia, nájsť efektívne taká tautológia x, že $C(x, y) = 0$ pre každé y?

6. Ak EF dokazuje efektívne $A \vee B$, dokazuje efektívne A alebo B?
   Ak áno a existujú tzv. super-bity, $\mathsf{lb}(\mathrm{SAT}, n^k)$ nemá krátke EF dôkazy.
   Má EF iné podobné konštruktívne vlastnosti?

7. Existuje generátor g ťažký pre všetky dôkazové systémy?
   T. j. existuje taká funkcia $g : \{0,1\}^n \mapsto \{0,1\}^{n+1}$ zobrazujúca binárne reťazce dĺžky n efektívne na binárne reťazce dĺžky $n+1$, že pre každý reťazec b dĺžky $n+1$ tvrdenie $\forall x_1, \ldots, x_n,\ b \neq g(x_1, \ldots, x_n)$ nemá krátky dôkaz v žiadnom dôkazovom systéme?

[6] Uvedené tvrdenia sa dajú efektívne vyjádriť ako tautológie za štandardných predpokladov z teórie zložitosti, pri pravdepodobnostných tvrdeniach sa použije aproximácia.

Obsah

Mathesis universalis
Ján Pich

2016
1. vydanie

Redaktor: Michal Rehúš
Jazyková úprava: Matúš Benkovič
Motív obalu: The7Dew

Vydavateľ: Literis, Čaklovská 2, 821 02 Bratislava
E-mail: literis@literis.sk

ISBN: 978-80-971502-2-8

ISBN 978-80-971502-2-8