## Test textu

ak ti $T$ v kontexte $K$ na pouzitie $P$ povie 1 resp 0 vezmi $f(T, K, P) = 1$ resp 0
takto sa pytaj kolko sa da v kolkych kontextoch to ide

$f(T_1, K_1, P_1) = 1/0$, ... , $f(T_i, K_j, P_k) = 1/0$ kde $P$ obsahuju text $S$
su vahy $S$ v korespondujucich kontextoch

najdi vahy howl v roku 1955
najdi vahy ariel v roku 1962
najdi vahy finnegans wake v roku 1939

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 $T$ v kontexte $K$ povie ze sa mylis vezmi $f(T, K, B$ je velka basen$) = 0$
a znovu najdi najefektivnejsiu $f$ ktora splnuje aj toto kriterium
ak $f(1, C, B$ je velka basen$) = 0$ uznaj ze si sa mylil
inak prehlas ze f je nejefektivnejsim 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 $X$

potom najdi texty $B$ tz $f(1, K, B$ je nedocenena basen$) = 1$
a $B$ tz $f(1, K, B$ je precenovana basen$) = 1$

ak ti povedia ze tomu nerozumies analyzuj text nerozumies tomu
a prehlas i follow the rules blindly

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

## Nasledovanie a narusovanie pravidiel

pre spravne nasledovanie pravidiel je potrebne spravne nasledovat pravidla ich nasledovania
preto je nakoniec nevyhnutne pravidla nasledovat slepo

apriori tak nie je mozne (vysvetlit ako) spravne nasledovat pravidla
kedze kazdy akt mozno interpretovat ako ich spravne nasledovanie

napriek tomu je mozne pravidla vysvetlit aposteriori
pretoze ich uz slepo nasledujeme

hypoteza: algoritmus interpretujuci pravidla najmensim obvodom konzistentnym s predoslymi skusenostami
bude pouzivat prirodzeny jazyk spravne

nech $M$ je (efektivny) proces rozhodujuci ci text $x$ splna kriteria $y$
a nech $S$ je (efektivny) sposob ako pre dane $y$ produkovat text $Sy$

ak pre kazde $S$ existuje $x$ a $y$ tz $M$ akceptuje $x$ $y$ no odmietne $Sy$ $y$
a navyse mozme toto zlyhanie $S$ efektivne dosvedcit
tj najst $x$ a $y$ pre dane $S$ efektivne

proces dosvedcujuci $x$ sa lysi od $S$-znamych sposobov produkovania textu pre kriteria $y$

hypoteza: efektivny algoritmus dosvedcujuci chyby znamych sposobov produkovania textu
produkuje zaujimave texty

## 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 kt 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

## Protokol efektivne rozpoznavajuci klamstvo velmi sebavedomej strany

input: $C$ tvrdi ze $y$ je najlepsia (znama) mozna odpoved
splnujuca iste efektivne overitelne kriteria
    (chceme overit ci ma $C$ pravdu

koduj tvrdenie "odpoved $x$ je lepsia nez $y$" ako SAT formulu
a tu reprezentuj ako multipremenny polynom $Y(x)$ stupna $d = n^{O(1)}$
kde $x = x_1, ..., x_n$ su premenne pre mozne odpovede dlzky $n$
    (chceme overit ci pre kazde 0/1 ohodnotenie $x$ plati $Y(x) = 0$
    (teda ci $\Sigma_{x \in 2^n} Y(x) \ mod \ p = 0$
    (kde $p$ je fixne provocislo z intervalu $(2^n, 2^{2n}]$

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

output: ak $C$ nezavaha ani po ohodnoteni vsetkych $x_1, ..., x_n$
    prehlas "$C$ je doverihodne" inak "$C$ je nedoverihodne"

{ak $y$ je najlepsia mozna odpoved, t.j. $\Sigma_{x \in 2^n} Y(x) = 0$
{    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 - doprovod k textom "Test textu", "Produkcia inovacii" a "Protokol efektivne rozpoznavajuci klamstvo velmi sebavedomej strany"

Je mozne pochopit a automatizovat vseobecne narocne procesy ako dokazovanie matematickych teorem ci dokonca pisanie poezie? Na tieto cinnosti dnes nemame efektivne algoritmy. Prekvapujuce vsak je, ze tiez nedokazeme vyvratit, ze by podobne algoritmy mohli existovat. Zmienene otazky pritom 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. Tu mozu tvorit povedzme 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 P mnoho dobrych vlastnosti na to, aby sa s nou pracovalo ako s aproximaciou problemov, ktore ide riesit efektivne, t.j. ktorych riesenie mozme v skutocnosti ocakavat bez toho, aby trvalo dlhsie nez povedzme predpokladany vek vesmiru. 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 su basne zmienene v nazve 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 dan tvrdenie (v praxi dokazatelna) teorema ma efektivne overitelne riesenie, ktorym je (kratky) dokaz daneho tvrdenia. Nadnesene sa da povedat, ze NP obsahuje vsetky problemy. Ak totiz mame problem, ktoreho riesenie nejde efektivne overit, mozme pochybovat o jeho zmysluplnosti.

Snad najdolezitejsim otvorenym problemom v teorii 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 napriklad 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 pisanie poezie?

Basen Test textu ilustruje algoritmus na pisanie poezie, ktory je "v tzv. polynomialnej hierarchii". Ak P=NP (ci lepsie povedane, 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 by tu znamenalo najst riesenie nejakej otazky, ktoru by dany algoritmus nevedel zodpovedat spravne. Z pohladu chybujuceho algoritmu by bolo take riesenie inovativnym textom (vymykajucim sa predoslym sposobom produkovania rieseni, ci specialne, poezie). V basni Produkcia inovacii je definovany 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 preverit, ci je moje presvedcenie, ze dalsi bit mojej basne ma byt 0 ci 1, spravne? Toto presvedcenie, ak nie je nahodne, sa zaklada na nejakej masinerii dovodov. Ak by som vedel rychlo overit jej doverihodnost, moja schopnost zachovat sa vzdy najlepsie ako mozem by bola podobne uzasna ako samotne spravne efektivne rozhodovanie dalsieho bitu mojej poezie. Basen Protokol efektivne.. popisuje taky test, ktory je aplikaciou znameho vysledku teorie zlozitosti, tzv. IP protokolu pre coNP problemy. Jeho nevyhodou je ale, ze vyzaduje, aby testovana masineria riesila prilis narocne problemy oznacovane ako #P. Adekvatnejsie by bolo testovanie, pri ktorom by sme neziadali, aby riesila viac nez to, co tvrdi, ze riesi. (Majuc tzv. compIP protokol pre coNP problemy by slo tuto masineriu otestovat s tym, ze by sme vyzadovali aby riesila nanajvys NP problemy, t.j. ak by tvrdila, ze riesi NP problemy, mohli by sme odhalit, ci ich naozaj riesi. Bohuzial vsak nie je zname, ci compIP protokol pre coNP problemy existuje.)

Hierarchiu problemov teorie zlozitosti naznacenu v predchadzajucom texte by slo rozvijat dalej. Jej najdolezitejsie 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 $C(x)$.*

Poznamka: 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.

Formalne mozme algoritmus z basne Produkcia Inovacii definovat nasledujuco.

**Definicia 2:** *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) formuli $x$ 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 $A$.*

# Appendix

## Formalizacia ultrafinitzmu

zmysel slov je dany ich pouzitim

pravidla pouzitia slov mozme nasledovat vzdy viacerymi nekonzistentnymi sposobmi

zmysel pravidiel ktory nakoniec pouzivame je ten najkrajsi. tym je urcene co je krasne ci efektivne

svet je totalita faktov

svet nasu skusenost mozme pouzit ako jazyk verejny a objektivny
ak hovorime o niecom co nejde vyjadrit vyjadrujeme presne to co hovorime
vyznam slov nieco nevyjadritelne neukazatelna skusenost bolest ktoru nik iny neciti.. je presne dany ich pouzivanim a nic viac neznamena

kazdu skusenost mozno nazvat jedinecnym slovom
slova pouzivane v suvislostiach v akych vystupuju korespondujuce skusenosti znamenaju to co tieto skusenosti

kazdy jazyk v ktorom dokazeme hovorit o pojmoch ako nekonecno je konecny. slova su prepisatelne do binarnych. tvrdeni je v kazdom okamihu konecne mnoho a su konecnej dlzky

objekt je konecny ak vieme prejst krok za krokom vsetky jeho prvky
ak by sme to dokazali zo skusenosti nekonecne krat vyznam slova konecne by sa zmenil

nasa skusenost s neustalym sa vynaranim novych veci je vyjadrena v pravidlach ako "pre kazde x existuje x+1". tie urcuju vyznam nekonecna
nemozeme prejst krok za krokom vsetky prvky nekonecna. ak by to boh dokazal porusil by logiku/pouzitie toho pojmu

mozme tvrdit ze existuje nekonecno ci nieco nevyjadritelne ale nebude to znamenat nic viac nez to co je dane konecnym mnozstvom konecnych vyjadreni

totalita faktov vsak neni len konecna je dosiahnutelna

tvrdit ze pre kazde x plati Tx znamena ze pre kazde x s ktorym mame skusenost plati Tx
ak napr nemame algoritmus pre nejaku ulohu tak (realne) neexistuje aj ked ho mozme zajtra najst

v matematike pouzivame kvantifikacie inak aj ked nieco realne neexistuje netvrdime ze to neexistuje kym to nevyvratime
kladieme doraz na minimalitu axiom kvoli (uspokojivejsiemu) vysvetleniu faktov (a jasnejsiemu vymedzeniu tych ktore sa zajtra zmenia)

je efektivne rozhodnutelne v akych suvislostiach su fakty pouzite napr kedy tvrdime ze su pravdive

aj ked je realne vsetko jasne hladame nove veci ako napr vysvetlenie faktov z minimalnych axiom

kazdy algoritmus rozhodujuci pravdivost tvrdenia v beznej mat teorii sa myli na istom explicitne danom vstupe ak je ta teoria konzistentna

mozme ale (efektivne) nachadzat dosiahnutelne dokazy tvrdeni?

existuje (efektivny) algoritmus ktory pre dane (efektivne overitelne) kriteria dokaze skonstruovat text ktory ich splnuje (ak taky text existuje)?

ukazat ze je problem lahky ma skutocne zmysel len konstrukciou algoritmu ktory ho riesi. teda dokaz existencie algoritmu ma skutocne zmysel len v teorii kde je dokaz existencie vzdy dosvedceny konstrukciou. v dosiahnutelnej matematike (fm)

ukazat obtiaznost problemu ma zmysel len konstrukciou algoritmu (efektivne) dosvedcujuceho chyby kazdeho potencialneho algoritmu pre tento problem

inak v skutocnosti nie sme schopni rozpoznat ze dany algoritmus nefunguje a prakticky tak moze byt problem lahky

fm pozostava z dokazov v akejkolvek formalizacii matematiky (zfc a rozsirenia) tz pre kazde dokazane tvrdenie typu existuje y A(x,y) existuje efektivna funkcia f tz A(x,f(x,n)) kde n je dlzka dokazu