Matematická logika

R. Lukoťka, J. Mazák

FMFI UK Bratislava

Zhrnutie: logika 1. rádu (FOL, niekedy tiež FO)

Uvažujme jazyk logiky 1. rádu a v ňom teóriu T a formulu F.

Theorem (Korektnosť logiky 1. rádu)

Ak $T \vdash F$, tak $T \models F$.

Theorem (Úplnosť logiky 1. rádu)

Ak $T \models F$, tak $T \vdash F$.

Theorem (Kompaktnosť logiky 1. rádu)

Ak je teória T nesplniteľná, existuje konečná podmnožina T, ktorá je nesplniteľná. // Ak každá konečná podmnožina T má model, tak aj T má model.

Reading: LogicStudyGuide 3.1, 3.2, 3.3 (cca 10 strán)

Many-sorted logic

V bežnej matematickej práci chceme kvantifikovať objekty rôznych typov osobitne, nepoužívame jednu spoločnú doménu, ale čiastkové (napr. pri axiomatizácii vektorových priestorov sú čísla a vektory rôzne druhy objektov). Toto možno vo FOL riešiť predikátmi pre jednotlivé druhy, čo vedie k neobratným formuláciám s implikáciami:

$$\forall a \forall x \forall y (S(a) \wedge V(x) \wedge V(y) \rightarrow a(x+y) = ax + ay)$$

Many-sorted logic

Je však možné upraviť FOL tak, že bude priamo v jazyku využívať objekty rôznych druhov s osobitnými doménami. Dostaneme tak *many-sorted logic*.

Všetky dôležité vlastnosti FOL sa zachovajú (úplnosť, kompaktnosť...). Cenou za šikovnejšie vyjadrovanie formúl sú o čosi zložitejšie/dlhšie dôkazy, preto sa v prácach o logike využíva bežná definícia FOL s jedinou doménou.

Reading: LogicStudyGuide 4.1 (2 strany)

Order-sorted logic

V many-sorted logic sú domény pre jednotlivé typy disjunktné. To sa pre modelovanie reálneho sveta neraz nehodí, môžeme preto zaviesť typovú hierarchiu, kde objekty jedného typu sú podmnožinou objektov iného typu (ako poznáme v objektových programovacích jazykoch). Získame tak *order-sorted logic*.

Toto je tiež možné preložiť do FOL, stačí zaviesť predikát P_i pre každý typ s_i a pridať axiómu $\forall x\,(P_i(x) \rightarrow P_j(x))$ pre každé dva typy také, že $s_i \subseteq s_j$.

Opačný postup (zavedenie typov namiesto predikátov arity 1) môže pomôcť pri automatizovanom dokazovaní.
Príklad: Ch. Walther 1985 (link)

Ako vyjadriť indukciu

Predstavme si, že chceme pridať axiómu umožňujúcu využívať v dôkazoch matematickú indukciu. Nech P^1 je predikát a S^1 funkčný symbol pre nasledovníka. Chceli by sme

$$\forall P \quad \Big(\big(P(0) \wedge \forall x\big(P(x) \to P(S(x))\big)\big) \to \forall x\, P(x)\Big).$$

Toto je však formula logiky 2. rádu — v logike 1. rádu nič ako $\forall P$ pre predikát P nemáme. Jediné, čo nám ostáva, je pridať axiómu

$$\big(P(0) \wedge \forall x\big(P(x) \to P(S(x))\big)\big) \to \forall x\, P(x),$$

kde P je predikát *vyjadriteľný formulou* v našom konkrétnom prvorádovom jazyku $\mathcal{L}$. To má nevýhody:

- dôkaz využívajúci túto axiómu bude platiť len pre jeden konkrétny predikát;
- dôkazy sa budú týkať len predikátov vyjadriteľných v $\mathcal{L}$.

Ako vyjadriť tranzitívny uzáver

Majme binárnu reláciu reprezentovanú predikátom R. Pre každé $n \geq 1$ označme $R_n(a, b)$ formulu

$$\exists x_1 \, \exists x_2 \cdots \exists x_n \big(R(a, x_1) \wedge R(x_1, x_2) \wedge \cdots \wedge R(x_n, b) \big)$$

(jednotlivé formuly R_n sú konečné, ale je ich nekonečne veľa — pre každé n by sme mohli náš jazyk rozšíriť o predikát, ktorý R_n definuje).
Ďalej nech $R_0 = R$ a nech

$$R^*(a, b) \quad \Leftrightarrow \quad \text{existuje } n \text{ také, že platí } R_n(a, b).$$

Naše R^* vyjadruje tranzitívny uzáver relácie R.
Dá sa však R^* vyjadriť prvorádovou formulou?

Ako vyjadriť tranzitívny uzáver

Predpokladajme, že R^* sa dá vyjadriť prvorádovou formulou. Uvažujme nekonečnú prvorádovú teóriu

$$T = \{\neg R_0(a,b), \neg R_1(a,b), \ldots, \neg R_n(a,b), \ldots, \quad R^*(a,b)\}.$$

- T nemá model: ak $R^*(a,b)$, tak pre nejaké n platí $R_n(a,b)$.
- Každá konečná podmnožina T má model: ak n je najväčší z indexov R_n obsiahnutých v podmnožine, vyhovujúci model je napr. $D = \mathbb{N}$, $i(R) = \{(0,1),(1,2),\ldots,(n+1,n+2)\}$, $i(a) = 0$, $i(b) = n+2$.

Tieto dve pozorovania sú však v spore s kompaktnosťou FOL. Formulou logiky 1. rádu tranzitívny uzáver nevieme vyjadriť.

Ako vyjadriť tranzitívny uzáver

V logike 2. rádu by to už šlo: $R^*(a,b) \leftrightarrow$

$$\forall P \Big(\Big(\forall x \left(R(a,x) \rightarrow P(x)\right) \wedge$$

$$\wedge \forall x \forall y \left(P(x) \wedge R(x,y) \rightarrow P(y)\right) \Big) \rightarrow P(b) \Big)$$

Modrej vlastnosti predikátu P hovoríme R-uzavretosť. Tento popis je založený na tom, že vlastnosť „*b je R-potomkom a*“ definujeme ako „*b* dedí od *a* každú vlastnosť, ktorú majú všetci priami *R*-potomkovia *a* a zároveň sa zachováva cez *R*“.

Logika 2. rádu

Syntax a sémantiku logiky 1. rádu možno rozšíriť o premenné pre predikáty, ktoré je dovolené kvantifikovať. Dostaneme tak logiku 2. rádu (SO).

Ak sa pri kvantifikovaní predikátov obmedzíme na predikáty arity 1, dostaneme monadickú logiku 2. rádu (MSO). Takéto predikáty de facto zodpovedajú množinám prvkov domény, čiže v MSO možno vyjadriť bežné veci z teórie grafov apod.

Použitie formúl 2. rádu zvyšuje vyjadrovaciu silu jazyka, ale oproti FOL stratíme kompaktnosť a úplnosť.

Reading: LogicStudyGuide 4.2 (5 pages)

Deskriptívna zložitosť

Pri klasickom pohľade na zložitosť algoritmov skúmame, koľko krokov spraví Turingov stroj v závislosti od veľkosti vstupu, resp. či Turingov stroj s dodatočnými obmedzeniami vôbec vie problém riešiť.

Alternatívny pohľad: akú zložitú logickú formulu potrebujeme na popis daného problému (jazyka akceptovaného Turingovým strojom)?

Deskriptívna zložitosť

Príklady rôzne zložitých formúl:

- prvorádová logika (FOL)
- druhorádová logika (SOL) – kvantifikujeme aj predikáty/množiny objektov
- existential SOL:
  $\exists X_1 \, \exists X_2 \cdots \exists X_n \;\; F, \qquad F$ je formula FOL
- universal SOL:
  $\forall X_1 \, \forall X_2 \cdots \forall X_n \;\; F, \qquad F$ je formula FOL

Deskriptívna zložitosť

Existencia trojuholníka v grafe sa dá vyjadriť vo FOL:

$$\exists x\, \exists y\, \exists z\ \big(V(x) \wedge V(y) \wedge V(z) \wedge \\ \wedge\, x \neq y \wedge y \neq z \wedge z \neq x \wedge E(x,y) \wedge E(y,z) \wedge E(z,x)\big)$$

Pri testovaní, či konkrétny konečný graf spĺňa túto formulu, nerozhodujeme o platnosti formuly vo všeobecnosti (to je nerozhodnuteľný problém), ale len vyhodnocujeme jej splnenie v konkrétnej interpretácii:
predikát V popisuje vrcholy, E hrany.

Deskriptívna zložitosť

Pri vyhodnotení kvantifikátora stačí prejsť cez všetky vrcholy, čiže pre k kvantifikátorov vo formule preveríme $O(|V|^k)$ možností.

Preto problém existencie trojuholníka v konečnom grafe (i iné popísateľné prvorádovou formulou) patrí do triedy P (zjemnenie úvah umožňuje dokázať príslušnosť do LOGSPACE).

Deskriptívna zložitosť

Existencia 3-farbenia grafu: NP-complete;
vyjadriteľné v existential SOL (nevieme, či vo FOL):

$$\exists R\, \exists G\, \exists B \big(\forall x\, \forall y\ (E(x,y) \rightarrow \neg R(x) \vee \neg R(y)) \wedge \cdots \big)$$

(R, G, B sú predikáty vyjadrujúce jednotlivé farby).

Existencia predikátu R zodpovedá existencii podmnožiny všetkých vrcholov; túto podmnožinu vieme nedeterministicky uhádnuť a vyhodnotenie prvorádovej časti formuly už pridá len polynomiálny faktor (exponent závisí od počtu vnorených kvantifikátorov), preto tento problém patrí do triedy NP.

Deskriptívna zložitosť

- FOL: trieda zložitosti AC^0 (vlastná podmnožina LOGSPACE)
- FOL + tranzitívny uzáver: non-deterministic LOGSPACE
- FOL + least fixed point operator: P
  (súvisí s databázami: dotazy sú prvorádové formuly a navyše je povolená rekurzia počítaná seminaivnou evaluáciou, čiže ako najmenší pevný bod)
- existential SOL: NP
- universal SOL: co-NP
- SOL: PH (obsahuje NP aj coNP; PH $\subsetneq$ PSPACE)
- SOL + least fixed point operator: EXPTIME

Nerozhodnuteľnosť platnosti formuly vo FOL

Theorem

Platnosť formuly v prvorádovej logike je nerozhodnuteľná.

Dôkaz: redukciou na problém zastavenia.
Ukážeme, ako by sme vedeli rozhodnúť, či daný Turingov stroj M zastaví, ak by sme vedeli rozhodovať platnosť prvorádových formúl.

Nerozhodnuteľnosť platnosti formuly vo FOL

Jazyk našej logiky bude obsahovať

- indivíduová konštanta ε pre prázdny reťazec;
- unárny funkčný symbol a^1 pre každé písmeno a v abecede;
- binárny predikát f_q pre každý stav q TS M.

Naša interpretácia tohto jazyka:

- $a(w)$ označuje reťazec aw;
- $f_q(x, y)$ indikuje, že M dosiahne na danom vstupe stav q, pričom na páske je reťazec $\overline{x}y$ (x v opačnom poradí) a hlava M je na prvom znaku y.

Nerozhodnuteľnosť platnosti formuly vo FOL

Krok výpočtu zachytáva formula

$$\forall x \, \forall y \quad f_q(x, a(y)) \to f_{q'}(b(x), y)$$

M prečíta z pásky a, zapíše b, prejde zo stavu q do stavu q' a posunie hlavu doprava. (Takúto formulu pridáme do teórie popisujúcej činnosť M pre každú dvojicu a, b.)
Pre pohyb hlavy doľava máme formulu

$$\forall x \, \forall y \quad f_q(c(x), a(y)) \to f_{q'}(x, c(b(y))).$$

Nerozhodnuteľnosť platnosti formuly vo FOL

Pre polohu hlavy na ľavom okraji pásky pridáme

$$\forall y \quad f_q(\varepsilon, a(y)) \to f_{q'}(\varepsilon, b(y))$$

(hlava sa nehýbe, len prepisuje znak na páske).
Podobne doriešime aj polohu na pravom okraji, keď sa hlava posunie na časť pásky, kam sa ešte nezapisovalo.
Konkrétne detaily závisia od uvažovaného variantu Turingovho stroja (páska môže byť obojstranne nekonečná apod.).

Nerozhodnuteľnosť platnosti formuly vo FOL

Začiatok výpočtu z počiatočného stavu q_0 na slove w popisuje formula $f_{q_0}(\varepsilon, w)$ a zastavenie stroja v (jedinom) akceptačnom stave q_{acc} vyjadruje formula

$$F_M = f_{q_0}(\varepsilon, w) \wedge T \rightarrow \exists x \, \exists y \;\; f_{q_{\mathrm{acc}}}(x, y),$$

kde T je konjunkcia implikácií popisujúcich povolené prechody TS M.

Nerozhodnuteľnosť platnosti formuly vo FOL

Každý akceptačný výpočet M vieme prerobiť na dôkaz F_M (stačí opakovane používať modus ponens na príslušné implikácie). Keďže prvorádová logika je korektná, tak ak M zastaví, F_M je platná formula.
Naopak, ak F_M je platná, tak je pravdivá v každej intepretácii (štruktúre), čiže aj v tej našej týkajúcej sa TS M. Pritom premisy F_M sú v nej splnené, preto musí byť splnený aj záver $\exists x \, \exists y \, f_{q_{\mathrm{acc}}}(x, y)$, takže M zastaví.

Čiastočná rozhodnuteľnosť platnosti formuly

Theorem

Platnosť formuly v prvorádovej logike (so spočítateľným jazykom) je čiastočne rozhodnuteľná.

Dôkaz: stačí enumerovať všetky dôkazy v danom jazyku.

Theorem

Nesplniteľnosť formuly v prvorádovej logike je nerozhodnuteľná a čiastočne rozhodnuteľná.

Dôkaz: F je nesplniteľná $\Leftrightarrow \neg F$ je platná.

Čiastočná rozhodnuteľnosť platnosti formuly

Theorem

Splniteľnosť v prvorádovej logike nie je ani čiastočne rozhodnuteľná.

Dôkaz: ak by sme splniteľnosť vedeli rozhodovať čiastočne, môžeme paralelne spustiť testovanie splniteľnosti aj nesplniteľnosti, a jeden z týchto výpočtov by musel skôr či neskôr skončiť, čím by sme vedeli rozhodovať nesplniteľnosť formuly, a to je spor.

Konečné a nekonečné modely

Theorem (Trakhtenbrot 1950)

Platnosť formuly v prvorádovej logike je nerozhodnuteľná, aj keď sa obmedzíme na konečné modely.
(Nevieme rozhodnúť, či je daná formula splnená v každom konečnom modeli.)

Dôkaz (nepreberali sme, len informačne):
redukciou na problém zastavenia.

Plán na zvyšok semestra

- dôkaz vety o úplnosti FOL
- formalizácia aritmetiky
- efektívna axiomatizácia
- dôkaz Gödelovej vety o neúplnosti (možno oboch)
- axiomatizácia teórie množín (ZFC)

Konečné a nekonečné modely

Pod veľkosťou modelu rozumieme veľkosť jeho domény. Pre nekonečné modely zvyčajne hovoríme o kardinalite miesto veľkosti.

V prvorádovej logike s rovnosťou:

1. Nájdite splniteľnú teóriu, ktorá nemá žiaden konečný model.
2. Nájdite splniteľnú formulu, ktorá má len konečné modely. Môže takáto formula existovať vo FOL bez rovnosti?
3. Môže existovať teória s konečným modelom veľkosti aspoň k pre každé k, ale bez nekonečného modelu?

Monadic FOL

Monadická prvorádová logika (MFOL) je prvorádová logika, v ktorej nemáme funkčné symboly (ani konštanty) a predikáty majú len jeden argument. (V takomto jazyku možno vyjadriť bežné sylogizmy a počas väčšiny 19. storočia sa verilo, že postačuje na formalizáciu uvažovania.)

Theorem (1915)

Splniteľnosť formuly v MFOL je rozhodnuteľná.

Ak pridáme čo len jeden predikát arity 2, stratíme rozhodnuteľnosť.

Tento príklad ilustruje, že zvoliť jazyk s neprimerane veľkou vyjadrovacou silou (napr. pre databázový systém) prináša algoritmické problémy, preto to nie je samozrejmá voľba.

Monadic FOL

Uvažujme formulu F, ktorá obsahuje n predikátov a najviac k vnorených kvantifikátorov.

Lema: Ak má F model, má aj model veľkosti najviac $k \cdot 2^n$.

Náčrt dôkazu: Nech D je doména existujúceho modelu. Priradíme každému prvku D charakteristiku (binárny vektor obsahujúci 1 bit vyjadrujúci pravdivosť pre každý možný atóm pre všetky jednotlivé predikáty vyskytujúce sa v F).

Monadic FOL

Novú doménu D' zostrojíme tak, že do nej pre každú možnú charakteristiku c dáme k prvkov D s charakteristikou c (ak ich je takých v D menej, zoberieme všetky). Interpretáciu predikátov ponecháme pôvodnú. V novom modeli sa nepokazí vyhodnocovanie všeobecných kvantifikátorov (keďže $D' \subseteq D$). Neovplyvní to však ani existenčné kvantifikátory: ak pre F existovali svedkovia v pôvodnej doméne D, nepotrebovali sme ich pri vyhodnocovaní platnosti F viac ako k, a k potenciálne rôznych svedkov máme naďalej v D'.

Dôkaz vety: podľa lemy ak má F model, tak má aj konečný model ohraničenej veľkosti. K danej doméne je len konečne veľa interpretácií. Stačí tak preveriť konečný počet štruktúr.