

Logika a počítání

- * Souvislost počítání a logického uvažování, jejich limitů.
- * Gödel: aritmetika nemůže být úplná (matematické dokazování není všemocné), Turing: zastavení se nedá rozhodnout (počítání není všemocné).

Formální definice logického uvažování: Logický systém

Logický systém matematicky přesně definuje

- * co je logických výrok: **syntaxe**,
- * jaký je jeho význam: **sémantika**,
- * co je logická argumentace: **dokazovací systém**.

Část I

Výroková logika (VL)

Syntaxe a sémantika výrokové logiky

- * Pro danou množinu výrokových proměnných \mathbb{X} , **syntaxe** formulí je dána gramatikou

$$\varphi \rightarrow X \mid \neg(\varphi) \mid (\varphi \wedge \varphi) \quad \text{kde } X \in \mathbb{X}$$

- * Další logické spojky a zkratky mohou být definovány jako syntaktický cukr:

$$\begin{aligned} \varphi \vee \psi &= \neg(\neg\varphi \wedge \neg\psi), \quad \varphi \rightarrow \psi = \varphi \vee \neg\psi, \quad \varphi \leftrightarrow \psi = (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi), \\ \varphi \text{ XOR } \psi &= \neg(\varphi \leftrightarrow \psi), \quad \varphi \text{ NAND } \psi = \neg(\varphi \wedge \psi), \quad 0 = X \wedge \neg X, \quad 1 = X \vee \neg X \end{aligned}$$

- * **Ohodnocení** $I : \mathbb{X} \rightarrow \{0, 1\}$ přiřazuje Booleovské hodnoty proměnným.

- * **Sémantika** formulí je dána relací **splňování**, $I \models \varphi$:

$$\begin{array}{lll} I \models X & \text{právě tehdy, když} & I(X) = 1 \\ I \models \neg(\varphi) & \text{právě tehdy, když} & I \not\models \varphi \\ I \models (\varphi \wedge \varphi') & \text{právě tehdy, když} & I \models \varphi \text{ a zároveň } I \models \varphi' \end{array}$$

Platnost a splnitelnost, důsledek, ekvivalence

- * φ je **platná** (*tautologie*, $\models \varphi$) právě když $I \models \varphi$ pro všechna ohodnocení I
- * φ je **nesplnitelná** (*kontradikce*) právě když $I \models \varphi$ pro žádné ohodnocení I
 - φ je **platná** právě když $\neg\varphi$ je **nesplnitelná**
 - φ je **splnitelná** právě když $\neg\varphi$ je **neplatná**
- * φ a ψ jsou **logicky ekvivalentní** ($\varphi \leftrightarrow \psi$) pokud je $\varphi \leftrightarrow \psi$ tautologie
- * ψ je **logickým důsledkem** ($\varphi \Rightarrow \psi$) φ pokud je $\varphi \rightarrow \psi$ tautologie

* *Schémata výrokových axiomů*

$$(A1) \quad A \rightarrow (B \rightarrow A)$$

$$(A2) \quad (A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$$

$$(A3) \quad (\neg B \rightarrow \neg A) \rightarrow ((\neg B \rightarrow A) \rightarrow B)$$

kde A, B, C jsou libovolné výrokové formule.

* *Odvozovací pravidlo modus ponens (pravidlo odloučení)*

(MP) Z předpokladů A a $(A \rightarrow B)$ odvodíme závěr B .

* (A1): Pokud A určitě platí a dovím se B , tak A bude pořád platit.

* (A2): V podstatě tranzitivita implikace.

* (A3): Pokud z výroku $(\neg B)$ plyne spor, pak neplatí. Tedy platí opak, B .

* (MP): Z (*prší \rightarrow nemám deštník*) a *prší* odvodím, že *nemám deštník*.

Důkaz formule φ (z množiny předpokladů P),

kde T je množina formulí,

je sekvencí formulí $\varphi_1, \dots, \varphi_n$, kde $\varphi_n = \varphi$ a pro každé $i : 1 \leq i \leq n$,

formule φ_i je axiomem (nebo prvkem P)

nebo vznikla z $\varphi_1, \dots, \varphi_{i-1}$ aplikací odvozovacích pravidel. Píšeme $\vdash \varphi$ ($P \vdash \varphi$).

Příklad důkazu: $\vdash A \rightarrow A$

- | | | |
|-----|---|------------|
| (1) | $A \rightarrow ((A \rightarrow A) \rightarrow A)$ | (A1) |
| (2) | $(A \rightarrow ((A \rightarrow A) \rightarrow A)) \rightarrow ((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A))$ | (A2) |
| (3) | $(A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A)$ | (1),(2) MP |
| (4) | $A \rightarrow (A \rightarrow A)$ | (A1) |
| (5) | $A \rightarrow A$ | (3),(4) MP |

Pozn.:

- (1) je (A1) pro volbu $A, A \rightarrow A$ za A, B
- (2) je (A2) pro volbu $A, A \rightarrow A, A$ za A, B, C
- (3) je závěr MP z předpokladů (1), (2)

Normální formy

- * **Literál** je buď proměnná (X) nebo negace proměnné ($\neg X$).
- * **Negační normální forma** (NNF)
 - pouze \wedge , \vee a \neg , \neg pouze v literálech.
 - převod do NNF odstraněním ostatních spojek a aplikací De Morganových zákonů.
- * **Disjunktivní normální forma** (DNF) je disjunkce konjunkcí literálů.
Konjunktivní normální forma (CNF) je konjunkce disjunkcí literálů.
 - převod do [C,D]NF převodem do NNF a použitím distributivních zákonů.

SAT – Splnitelnost formulí výrokové logiky

- * SAT = {formule VL φ | φ je splnitelná} je rozhodnutelný problém.
- * Složitost známých algoritmů v nejhorším případě více než $2^{|\mathbb{X}|}$. NP-úplný problém ... (Vyzkoušej všechna možná ohodnocení)
- * SAT solvery rychlé v praktických případech (zvládají stovky tisíc proměnných).
 - pracují s DNF
 - základ backtracking, 30+ let heuristik a chytrosti (SAT competition)
- * Aplikace v průmyslu:
Plánování, kombinatorická optimalizace, lámání šifer, analýza software a hardware, generování testů, automatický design, optimalizace kompilace, zpracování přirozeného jazyka, herní AI, biologie, zdravotnictví, ...

SAT a normální formy

- * Převod do ekvivalentní CNF je exponenciální.
- * SAT solver může pracovat s CNF díky Tseitinově transformaci do CNF, která není ekvivalentní, ale ekvisplnitelná:
 - Induktivně ke struktuře formule nahrazuj všechny podform. φ za nové proměnné x_φ a přidávej nové konjunkty ($x_\varphi \leftrightarrow \varphi$).
 - Pro poslední podformuli ψ přidej konjunkt x_ψ .
 - Převeď nové konjunkty do CNF (NNF a distributivní zákony).
- * Co by znamenala existence efektivního polynomiálního překladu do DNF?

Část II

Predikátová logika (PL)

Syntaxe predikátové logiky

- * **Signatura** jazyka PL dvojice $\langle \mathcal{F}, \mathcal{P} \rangle$ množin **funkčních a predikátových symbolů**, každý asociovaný a **aritou** z \mathbb{N} . Píšeme $f/n \in \mathcal{F}$, $p/n \in \mathcal{P}$ ve smyslu symbol s ar. n . (sig. je tedy vlastně trojice $\langle \mathcal{F}, \mathcal{P}, ar \rangle$ kde $ar : \mathcal{F} \cup \mathcal{P} \rightarrow \mathbb{N}$ je arita)
- * **Jazyk formulí PL** se signaturou $\langle \mathcal{F}, \mathcal{P} \rangle$ a množinou proměnných \mathbb{X} je dán gramatikou

$$\begin{aligned} \varphi &\rightarrow p(\overbrace{t, \dots, t}^n) \mid \neg(\varphi) \mid (\varphi \wedge \varphi) \mid (\exists x\varphi) && p/n \in \mathcal{P} \\ t &\rightarrow f(\underbrace{t, \dots, t}_m) \mid x && x \in \mathbb{X}, f/m \in \mathcal{F} \end{aligned}$$

- * t je **term**
- * $p(t, \dots, t)$ je **atomická formule**
- * Další spojky a 0 a 1 jsou syntaktický cukr jako ve VL, plus $(\forall x\varphi) = \neg(\exists x\neg\varphi)$.
- * **Volný/vázaný výskyt proměnné** ..., **věta** = uzavřená formule (bez volných prom.)

Sémantika predikátové logiky

- * **Interpretace/realizace** jazyka se sig. $\langle \mathcal{F}, \mathcal{P} \rangle$ a proměnnými \mathbb{X} je pár $I = (D_I, \alpha_I)$, kde
 - D_I je **doména**, libovolná neprázdná množina
 - α_I přiřazuje prvky D_I a relace/funkce nad D_I proměnným a pred./fun. symbolům.
- * Termy jsou interpretovány jako hodnoty z domény:

$$\alpha_I(f(x_1, \dots, x_n)) = \alpha_I(f)(\alpha_I(x_1), \dots, \alpha_I(x_n)) \quad f/n \in \mathcal{F}$$

- * Sémantika formulí je dána relací splnění v interpretaci, $I \models \varphi$:

$$\begin{array}{ll} I \models p(t_1, \dots, t_n) & \text{právě když } (\alpha_I(t_1), \dots, \alpha_I(t_n)) \in \alpha_I(p) \quad p/n \in \mathcal{P} \\ I \models \varphi_1 \wedge \varphi_2 & \text{právě když } I \models \varphi_1 \text{ a zároveň } I \models \varphi_2 \\ I \models \neg \varphi & \text{právě když } I \not\models \varphi \\ I \models \exists x \varphi & \text{právě když } I[x/v]^* \models \varphi \text{ pro nějaké } v \in D_I \end{array}$$

Pokud $I \models \varphi$, potom I je **modelem** φ .

* $f[x/v]$ vznikne z f přeměnováním funkční hodnoty $f(x)$ na v .

† Místo $\alpha_I(x/f/p)$ někdy píšeme $I(x/f/p)$.

* Axiomy VL a modus ponens.

* **Schéma axiomů kvantifikátoru:** Není-li x volné ve φ , pak

$$(\forall x(\varphi \rightarrow \psi)) \rightarrow (\varphi \rightarrow (\forall x\psi))$$

* **Schéma axiomů substituce:** pokud t je term substituovatelný^a za x do φ

$$(\forall x \varphi) \rightarrow \varphi[x/t]$$

* **Axiomy rovnosti:** Pro $f_{/n} \in \mathcal{F}$, $p_{/n} \in \mathcal{P}$ a proměnné $x, x_i, y_i, 1 \leq i \leq n$

$$x = x, \quad x_1 = y_1 \rightarrow (x_2 = y_2 \rightarrow (\dots (x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)) \dots))$$

$$x_1 = y_1 \rightarrow (x_2 = y_2 \rightarrow (\dots (x_n = y_n \rightarrow p(x_1, \dots, x_n) \rightarrow p(y_1, \dots, y_n)) \dots))$$

* **Pravidlo zobecnění (generalizace):**

Z předpokladu φ odvodíme závěr $(\forall x\varphi)$.

^a t neobsahuje volné proměnné φ

Příklady axiomů PL

* axiom kvantifikátoru: $(\forall x(\varphi \rightarrow \psi)) \rightarrow (\varphi \rightarrow (\forall x\psi))$

$(\forall \text{Brňák} (\text{je hezky} \rightarrow \text{Brňák je na přehradě})) \rightarrow$
 $(\text{je hezky} \rightarrow (\forall \text{Brňák}(\text{Brňák je na přehradě})))$

* axiom substitute: $(\forall x \varphi) \rightarrow \varphi[x/t]$

$(\forall \check{c} \text{ zaslouží_soucít}(\check{c})) \rightarrow \text{zaslouží_soucít}(\text{matka}(\text{manželka}(x)))$

* ax. rovnosti: $x_1 = y_1 \rightarrow (\dots (x_n = y_n \rightarrow p(x_1, \dots, x_n) \rightarrow p(y_1, \dots, y_n)) \dots)$

$(\text{pivo} = \text{chleba} \rightarrow (\text{snídaně}(\text{chleba}, \text{máslo}) \rightarrow \text{snídaně}(\text{pivo}, \text{máslo})))$

Příklad důkazu v PL: $p(x, y) \vdash p(y, x)$

(1)	$p(x, y)$	(předpoklad)
(2)	$\forall y p(x, y)$	(pravidlo zobecnění)
(3)	$\forall x \forall y p(x, y)$	(pravidlo zobecnění)
(4)	$(\forall x \forall y p(x, y)) \rightarrow (\forall y p(z, y))$	(axiom substituce, x za z)
(5)	$\forall y p(z, y)$	(MP)
(6)	$(\forall y p(z, y)) \rightarrow p(z, x)$	(axiom substituce, y za x)
(7)	$p(z, x)$	(MP)
(8)	$(\forall z p(z, x))$	(pravidlo zobecnění)
(9)	$(\forall z p(z, x)) \rightarrow p(y, x)$	(axiom substituce, z za y)
(10)	$p(y, x)$	(MP)

Pozn.:

- (1): předpoklad; (2): p. zobec. pro $\varphi = (1)$, $x = y$; (3): p. zobec. pro $\varphi = (2)$;
(4): ax. subst. pro $\varphi = (3)$ a $t = z$; (5): MP pro $\varphi = (3)$ a $\varphi \rightarrow \psi = (4)$;
(6): ax. subst. pro $\varphi = (5)$, $x = y$ a $t = x$; (7): MP pro $\varphi = (5)$ a $\varphi \rightarrow \psi = (6)$;
(8): ax. subst. pro $\varphi = (7)$, $x = z$; (9): ax. subst. pro $\varphi = (8)$, $x = z$ a $t = y$;
(10): MP pro $\varphi = (8)$ a $\varphi \rightarrow \psi = (9)$;

Část III

Vlastnosti logických systémů

Logický systém je efektivní, pokud můžeme efektivně ověřit korektnost logického argumentu, důkazu. TS ověří, že daný řetězec symbolů je důkaz.

Výroková i predikátová logika je efektivní.

* Protože můžeme algoritmicky ověřit

1. co je dobře formulovaná formule,
2. co je axiom,
3. že formule v důkazu byla odvozena odvozovacími pravidly z předchozích.

Dokazatelnost

versus

Platnost

$\vdash \varphi$

Dokazatelná (z axiomů pomocí odvoz. prav.).
Čistě syntaktická manipulace.

$\models \varphi$

Platná ve všech interpretacích.
Rozhoduje sémantika, význam.

Ve VL ověříme pravd'. tabulkou, SAT, ...
V PL problém s nekón. dom. a $\forall\psi$

$\vdash \varphi$ je ověřením $\models \varphi$
Důkaz je ověřením platnosti.

Mělo byt tedy platit, že

$\vdash \varphi \iff \models \varphi.$

Korektnost a sémantická úplnost

System je *korektní* pokud

$$\vdash \varphi \implies \models \varphi$$

Co je dokazatelné, to je platné.
Nemůžeme dokázat nesmysly.

System je sémanticky *úplný* pokud

$$\models \varphi \implies \vdash \varphi$$

Vše platné můžeme dokázat.

Korektnost a sémantická úplnost Hilbertovského systému pro VL a PL

Výroková i predikátová logika je korektní a sémanticky úplná. (Post, Gödel)

Pro libovolnou formuli VL nebo PL platí $\models \varphi \iff \vdash \varphi$.

Pro PL je to Gödelova věta o úplnosti.

Kurt Gödel



Brno, 1906 - 1978

Část IV

Teorie v PL

Platnost a dokazování ve vybraných strukturách

$$\not\models 1 + 1 = 2$$

Chceme vyjádřit a dokázat, že $1 + 1 = 2$ je platná v přirozených číslech, jak je známe (t.j. když 1, 2, a + jsou interpretovány, jak jsme zvyklí)

K tomu slouží logické teorie:

$$[\text{Teorie celých čísel}] \models 1 + 1 = 2$$

- * **Teorie** s jazykem L je množina T *speciálních axiomů*, vět z jazyka L .
- * **Model** T je interpretace \mathcal{M} jazyka L , kde $\mathcal{M} \models \psi$ pro všechny $\psi \in T$. $\mathcal{M} \models T$.
- * **Důsledek teorie** je formule φ platná ve všech jejích modelech. $T \models \varphi$.
- * **Dokazatelnost v teorii** T je dokazatelnost z předpokladů T . $T \vdash \varphi$

- * T -splnitelnost – formuli splňuje nějaký model teorie T
- * T -ekvivalence – formule mají stejné modely teorie T ,
- * T -důsledek ...

Stále platí Gödelova věta o úplnosti PL:

$$T \models \varphi \iff T \vdash \varphi$$

Příklady teorií

Teorie uspořádání T_{\leq} má speciální axiomy

$$\begin{aligned}\forall x(x \leq x) & \quad (\text{reflexivita}) \\ \forall x \forall y \forall z ((x \leq y \wedge y \leq z) \rightarrow x \leq z) & \quad (\text{tranzitivita}) \\ \forall x \forall y \forall z ((x \leq y \wedge y \leq x) \rightarrow x = y) & \quad (\text{antisymetrie})\end{aligned}$$

Příklad modelu: $D_{\mathcal{M}} = \{1, 2, 3\}$, $\mathcal{M}(\leq) = \{(1, 2), (1, 1), (2, 2), (3, 3)\}$, $\mathcal{M} \models T_{\leq}$
 $T_{\leq} \models x \not\leq y \vee y \not\leq x \vee x = y$ a také $T_{\leq} \vdash x \not\leq y \vee y \not\leq x \vee x = y$

Teorie grup T_G má speciální axiomy

$$\begin{aligned}\forall x \forall y \forall z (x \cdot (y \cdot z) = (x \cdot y) \cdot z) & \quad (\text{asociativita}) \\ \forall x (x \cdot e = x \wedge e \cdot x = x) & \quad (\text{neutrální prvek}) \\ \forall x \exists y (x \cdot y = e \wedge y \cdot x = e) & \quad (\text{inverzní prvky})\end{aligned}$$

$D_{\mathcal{M}} = \{0, 1\}$, $\mathcal{M}(e) = 1$, $\mathcal{M}(\cdot)$ = násobení ve zbytkové třídě 2
 $T_G \models x = e \rightarrow x \cdot x = x$ a také $T_G \vdash x = e \rightarrow x \cdot x = x$.

Příklady teorií: Peanova aritmetika T_{PA}

- $\forall x \neg(S(x) = 0)$ (nula je první)
- $\forall xy(S(x) = S(y) \rightarrow x = y)$ (každý má jiného následníka)
- pro formule φ jazyka T_{PA} s jednou volnou proměnou:
$$[\varphi(0) \wedge (\forall x(\varphi(x) \rightarrow \varphi(S(x))))] \rightarrow \forall x (\varphi(x))$$
 (ax. indukce)
- $\forall x(x + 0 = x)$ (0 je neutrální k +)
- $\forall xy(x + S(y) = S(x + y))$ (def sčítání)
- $\forall x(x \cdot 0 = 0)$ (0 je nulová k ·)
- $\forall xy(x \cdot S(y) = x \cdot y + x)$ (def násobení)

$0, S^1(0), S^2(S^1(0)), S^3(S^2(S^1(0))), S^4(S^3(S^2(S^1(0))))), S^5(S^4(S^3(S^2(S^1(0))))), \dots$

$$T_{PA} \models S^1(0) + S^1(0) = S^2(S^1(0)) \quad \text{a} \quad T_{PA} \vdash S^1(0) + S^1(0) = S^2(S^1(0))$$

Teorie racionálních čísel se sčítáním

1. $\forall x \forall y ((x \leq y \wedge y \leq x) \rightarrow x = y)$ (antisymmetry)
2. $\forall x \forall y \forall z ((x \leq y \wedge y \leq z) \rightarrow x \leq z)$ (transitivity)
3. $\forall x \forall y (x \leq y \vee y \leq x)$ (totality)
4. $\forall x \forall y \forall z ((x + y) + z = x + (y + z))$ (+ associativity)
5. $\forall x (x + 0 = x)$ (+ identity)
6. $\forall x (x + (-x) = 0)$ (+ inverse)
7. $\forall x \forall y (x + y = y + x)$ (+ commutativity)
8. $\forall x \forall y \forall z (x \leq y \rightarrow x + z \leq y + z)$ (+ ordered)
9. for each positive integer n ,
 $\forall x (nx = 0 \rightarrow x = 0)$ (torsion-free)
 $\forall x \exists y (x = ny)$ (divisible)

*where nx denotes $\overbrace{x + \dots + x}^n$

Teorie T je **bezesporná**, pokud neexistuje formule φ taková, že $T \vdash \varphi$ a $T \vdash \neg\varphi$.

- * Bezespornost je základ smysluplnosti. Axiomy si nesmí protiředit.
- * Ve sporné teorii je možné dokázat cokoliv.
Pokud $T \vdash \varphi$ a $T \vdash \neg\varphi$, pak i $T \vdash \varphi \wedge \neg\varphi$. Nepravda, 0, je tedy dokazatelná.
Z nepravdy pak plyne cokoliv (protože $0 \rightarrow \psi$ vždy platí).
- * Sporná teorie nemá žádný model, protože 0 nemůže platit v žádné interpretaci.

Teorie je bezesporná právě tehdy, když má model.

Bezespornost, příklad

Teorie T :

$$\begin{aligned} &\forall x \forall y (x < y) \\ &\neg \forall x \forall y (x < y) \\ &\forall x (x \not< x) \end{aligned}$$

Teorie T' :

$$\begin{aligned} &\forall x \forall y \forall z (x.(y.z)) = ((x.y).z) \\ &\forall x (x.e = e.x = x) \\ &\exists z (z \neq e \wedge \forall x (x.z = z.x = x)) \end{aligned}$$

Teorie T'' :

$$\begin{aligned} &\exists x (f(x) = 0) \\ &\forall x \forall y (f(x) = f(y)) \\ &\neg \exists x (f(x) = x) \end{aligned}$$

Russelův paradox ve Fregeho naivní teorii množin

Gottlob
Frege



Bertrand Russel



Mějme množinu M , obsahující všechny množiny, které nejsou svým vlastním prvkem.

Je M svým vlastním prvkem?

Jestli je M vlastním prvkem, pak, podle definice, není vlastním prvkem.

Jestli M není vlastním prvkem, pak, podle definice, je vlastním prvkem.

Zatr sakr ...

Syntaktická úplnost teorií, intuitivně

- * Úplnost teorie formalizuje „přesnost“, nebo „jednoznačnost“ definice.
- * Definice je přesná, jednoznačná, pokud definuje přesně jednu věc, jednu mat strukturu. Nevyhovuje jí hned několik různých mat. struktur.

Syntaktická úplnost teorií

Teorie T je **úplná**, pokud pro každou větu φ jazyka T platí
 $T \vdash \varphi$, nebo $T \vdash \neg\varphi$.
(Každou uzav. formuli jazyka je v T možno dokázat nebo vyvrátit.)

- * Nechť T má právě jeden model \mathcal{M} (až na izomorfismus).
- * Pro každou formuli jazyka T máme $\mathcal{M} \models \varphi$, nebo $\mathcal{M} \models \neg\varphi$ (z def. $\mathcal{M} \models$).
- * Protože \mathcal{M} je jediný, tak $T \models \varphi$, nebo $T \models \neg\varphi$.
- * Z věty o úplnosti PL ($T \models \psi \Leftrightarrow T \vdash \psi$) plyne, že $T \vdash \varphi$, nebo $T \vdash \neg\varphi$.

Teorie má jediný model \Rightarrow je úplná.

Je úplná \Leftrightarrow nemá dva modely rozlišitelné formulemi jazyka
(t.j., neexistují $\mathcal{M}, \mathcal{M}', \varphi$ tak, že $\mathcal{M} \models \varphi$ a $\mathcal{M}' \models \neg\varphi$)

Syntaktická úplnost teorií, příklady

Teorie relací ekvivalence:

$$\begin{aligned} & \forall x (x \sim x) \\ & \forall x \forall y (x \sim y \rightarrow y \sim x) \\ & \forall x \forall y \forall z ((x \sim y \wedge y \sim z) \rightarrow x \sim z) \end{aligned}$$

- * $\mathcal{M}: D_{\mathcal{M}} = \{1\}, \mathcal{M}(\sim) = \{(1, 1)\}$
- * $\mathcal{M}': D_{\mathcal{M}'} = \{1, 2\}, \mathcal{M}'(\sim) = \{(1, 1), (2, 2)\}$
- * $\varphi: \forall x \forall y (x \sim y), \mathcal{M} \models \varphi$ a $\mathcal{M}' \models \neg \varphi$, tedy není úplná.

Chceme definovat dvouprvkový svaz s maximem 1 a minimem 0.

$$\begin{aligned} & 0 \leq 1 \wedge 0 \leq 0 \wedge 1 \leq 1 \wedge 1 \not\leq 0 \\ & \forall x (x = 0 \vee x = 1) \end{aligned}$$

$\mathcal{M}: D_{\mathcal{M}} = \{a, b\}, \mathcal{M}(0) = a, \mathcal{M}(1) = b, \mathcal{M}(\leq) = \{(a, b), (a, a), (b, b)\}$
Jediný model \Rightarrow je úplná.

Příklad (beze)sporné a (ne)úplné teorie

$$\varphi_1 : \forall y(1.y = y) \quad \varphi_2 : \forall y(1.y = 1) \quad \varphi_3 : \neg\exists x(x.x = x)$$

* $T_1 = \{\varphi_1\}$

- bezesporná, má model \mathcal{M} s $D_{\mathcal{M}} = \{a\}$, $\mathcal{M}(1) = a$, a $\mathcal{M}(\cdot) = \{((a, a), a)\}$
- není úplná, má model \mathcal{M}' s $D_{\mathcal{M}'} = \{a, b\}$, $\mathcal{M}'(1) = a$,
 $\mathcal{M}'(\cdot) = \{((a, a), a), ((a, b), b), ((b, a), a), ((b, b), a)\}$
a pro $\psi = \exists x(x.x \neq x)$ je $\mathcal{M} \not\models \neg\psi$ ale $\mathcal{M}' \models \psi$.

* $T_2 = \{\varphi_1, \varphi_2\}$

- stále bezesporná (\mathcal{M})
- úplná, \mathcal{M} je jediný (každá φ v \mathcal{M} platí pozitivní nebo negovaná, a tedy platí pro všechny modely, což se pak z Gö. v. o ú. dá dokázat)

* $T_3 = \{\varphi_1, \varphi_3\}$

- sporná, obojí nemůže platit zároveň v žádné interpretaci, nemá model
- každá sporná teorie je i úplná, protože ze sporu je dokazatelné vše

Efektivnost + korektnost + synt. úplnost \Rightarrow rozhodování důsledků teorie

Pro efekt. úplnou teorii T ex. program/TS *Rozhodovač důsledků T /platnosti v T .*

Pro danou φ pracuje takto:

foreach slovo w z abecedy jazyka T // slova v lexikografickém pořadí

do

if w je důkaz φ v T **then** φ je důsledkem T

if w je důkaz $\neg\varphi$ **then** φ není důsledkem T

- * Efektivnost zaručuje, že jsme schopni ověřit, co je důkaz.
- * Korektnost zaručuje, že odpověď je správná.
- * Synt. úplnost zaručuje, že program skončí nalezením důkazu nebo vyvrácení φ (jeden z nich existuje).

Příklad běhu Mecha. matematika: Je $T_{PA} \models S(0) + S(0) = S(S(0))$?

<i>a</i>	
<i>b</i>	$xyp \Rightarrow$
<i>c</i>	...
...	$x = y \rightarrow p$
\wedge	...
\neg	$xy \Rightarrow (((, ((\wedge \neg \neg x, , \exists$
\exists	...
<i>aa</i>	$p \rightarrow ((p \rightarrow p) \rightarrow p), (p \rightarrow (p \rightarrow p) \rightarrow (p \rightarrow p)), \forall x \forall y x=y$
<i>ab</i>	...
<i>ac</i>	Psal jsem si se slepýšem, ale už si nepíšem. (Plíhal)
...	...
<i>aab</i>	$p \rightarrow ((p \rightarrow p) \rightarrow p), (p \rightarrow ((p \rightarrow p) \rightarrow p) \rightarrow ((p \rightarrow (p \rightarrow p)) \rightarrow (p \rightarrow p))), \underline{(p \rightarrow (p \rightarrow p) \rightarrow (p \rightarrow p))}$
...	...
<i>ab</i> \exists	...
...	...
<i>a</i> \vee <i>b</i> \neg ($\forall \neg (S(x)=0) \dots \dots \dots \text{důkaz} \dots \dots \dots, \underline{S(0)+S(0)=S(S(0))}$
...	

Teorie je (částečně) rozhodnutelná, pokud $\{\varphi \mid T \models \varphi\}$ je (částečně) rozhodnutelná.

- * Pro efektivní, bezspornou a úplnou teorii je $\{\varphi \mid T \vdash \varphi\} = \{\varphi \mid T \models \varphi\}$ rozhodnutelná (generátor časem vygeneruje důkaz φ nebo $\neg\varphi$)

Efektivní, bezsporná a úplná teorie je rozhodnutelná.

Efektivní a bezsporná teorie je částečně rozhodnutelná.

- * Sporná teorie je rozhodnutelná triviálně, protože každá formule je jejím důsledkem.
- * Rozhodnutelnost neimplikuje úplnost!
- * Formule φ s volnými proměnnými x_1, \dots, x_n je T -splnitelná $\Leftrightarrow T \models \exists x_1 \dots \exists x_n \varphi$.
- * Věta φ je T -splnitelná $\Leftrightarrow T \models \varphi$.

Část V

SMT-solving

Co a k čemu je SMT-solving?

- * **SMT - SAT Modulo Theory**. SAT solver, s nadstavbou pro prvořádové teorie.
- * Omezený ale praktický mechanický matematik (nemá rád kvantifikátory, občas neví, ...).
- * Rozhodne, zda je daná formule v dané teorii splnitelná, příp. vrátí model.

$(x + 5 \leq 6 \wedge x.y = z + 2 \wedge x/5 = z) \rightarrow (z \bmod y = 10)$ (aritmetika na $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$)

$a[i]_r = e \rightarrow \forall j((a[i, e]_w)[j]_r = a[j]_r)$ (teorie polí)

$((y_4 + 8 \ 1) \not\prec_8 y_3) \wedge y_2 = y_4 + 8 y_1$ (teorie bitových vektorů)

$x \in \text{Name}: [a - z]^* \wedge y \in \text{Year}: .* \wedge z = x.y \wedge |z| \leq 10$ (teorie řetězců)

- * Vyřeš sudoku! Co naskládat do kamionu, aby to v rámci váhového limitu mělo co největší cenu? Naplánuj cestu. Navrhni rozvrh. Jak donutit program projít danými řádky? Je toto invariant cyklu? ...
- * Aplikace v automatickém testování, verifikaci, v návrhu a syntéze systémů, plánování, optimalizaci, inferenci typů, AI, ...
 - Certora, Trail of Bits – verifikace block-chain kontraktů (Brno)
 - Honeywell – automatické generování testů (Brno)
 - RedHat – pokusy o formální metody (Brno)
 - Amazon – analýza konfig. Cloud služeb (N. Rungta: A Billion SMT Queries per Day)
 - Microsoft – formální metody obecně
 - AI (ChatGPT + SMT?)

Architektura SMT-solveru

Vstup:

- * formule
- * identifikace teorie

Komponenty:

- * **SAT-solver**. Řeší výrokové formule v CNF, zvládá řádově až 10^7 proměnných.
- * **Theory-solver**. Hledá řešení *konjunkcí* v dané teorii.
Každá teorie potřebuje vlastní specializovaný algoritmus.

1. SAT-solverem najdi model Booleovské kostry, pokud neexistuje, vrať UNSAT.
2. Theory-solverem zkontroluj model z hlediska teorie.
 - Pokud je model splnitelný z hlediska teorie, vrať SAT,
 - jinak přidej Booleovské kostře negaci modelu jako naučenou klauzuli, goto 1.

Příklad

$$\varphi : (x < 0 \vee x \neq 2) \wedge (x < 2 \vee x = 3) \wedge (x = -1 \vee \neg x < 0), T_{\mathbb{Z}}$$

Bool. kostra: $(a \vee \neg b) \wedge (b \vee c) \wedge (d \vee \neg a)$

1. SAT-solver: SAT, $\{a \mapsto 1, b \mapsto 1, c \mapsto 1, d \mapsto 1\}$
Theory-solver: $(x < 0 \wedge x < 2 \wedge x = -1 \wedge x = 3)$ je UNSAT. Learned cl. $\neg(a \wedge b \wedge c \wedge d)$
2. SAT-solver: SAT, $\{a \mapsto 0, b \mapsto 1, c \mapsto 1, d \mapsto 0\}$
Theory-solver: $(\neg x < 0 \wedge x < 2 \wedge x = -1 \wedge \neg x = 3)$ je UNSAT. L. cl. $\neg(\neg a \wedge b \wedge c \wedge \neg d)$
3. SAT-solver: SAT, $\{a \mapsto 0, b \mapsto 0, c \mapsto 0, d \mapsto 1\}$
Theory-solver: $(\neg x < 0 \wedge \neg x < 2 \wedge \neg x = -1 \wedge x = 3)$ je SAT, $\{x \mapsto 3\}$ je řešení.