Abstrakt

Text je určen jako doplňkový k přednášce Matematická logika a Paradigmata programování 4.

Matematická logika - poznámky k přednáškám

Radim Bělohlávek

18. dubna 2006

1 Co je (matematická) logika?

Co je logika? Logika je vědou o správném usuzování. V logice jde o to, aby usuzování mělo správnou *formu* bez ohledu na *obsah*. Uvažujeme-li např. tvrzení "prší" a "jestliže prší, (pak) jsou silnice mokré", pak z nich (intuitivně zcela samozřejmě) lze odvodit tvrzení "silnice jsou mokré". Uvažujme jinou dvojici tvrzení, např. "Petr má hlad" a "Jestliže má Petr hlad, (pak) se (Petr) snaží sehnat něco k jídlu". Z těchto tvrzení plyne tvrzení "Petr se snaží sehnat něco k jídlu". V uvedených příkladech vyplývalo z dvojice tvrzení další tvrzení. Uvedené dvojice tvrzení měly zcela jistě jiný obsah ("prší" znamená zcela jistě něco jiného než "Petr má hlad"). Způsob, jakým jsme odvodili třetí tvrzení byl však v obou případech stejný; říkáme, že usuzování mělo stejnou formu. Tuto formu je možné znázornit *symbolicky* takto: z tvrzení, které má formu "A", a tvrzení, které má formu "$A \Rightarrow B$" ("jestliže A, pak B"), plyne tvrzení "B".

Uvedené rysy jsou pro moderní logiku charakteristické, proto je zopakujme: logika studuje formy usuzování bez ohledu na obsah a má (proto) symbolický charakter (jednotlivá tvrzení označujeme symboly-písmeny, např. výše uvedené symboly A a B; způsoby spojení tvrzení ve složitější tvrzení označujeme symboly, např. výše uvedené "$\Rightarrow$"). Pro uvedené rysy bývá moderní logika označována jako logika formální, popř. symbolická. Je pochopitelné, že symbolický charakter umožňuje logice snadněji odhlédnout od obsahu a soustředit se na formy usuzování.

Co je matematická logika? Učiníme-li studium forem usuzování předmětem našeho zájmu, vzniká přirozeně otázka, jakých metod při tomto studiu používat. Rozhodneme-li se používat metod matematiky, hovoříme o *matematické logice.* Studiem logiky matematickými prostředky lze dosáhnout hlubokých výsledků, některé ukážeme v tomto textu. V našem textu budeme pojednávat právě o matematické logice.

Logika: matematická a nematematická? Dospěli jsme ke stručné charakteristice toho, co nás bude zajímat především, tedy matematické logiky. Otázkou nyní zůstává, co dalšího kromě matematické logiky se ve světě logiky skrývá. Někdy bývá uváděno, že logika se dělí na matematickou logiku a *filozofickou logiku.*

Toto dělení má své historické důvody, které zde nemůžeme podrobně rozvádět. V současné době, zdá se, je však toto rozdělení umělé, nepřirozené a překonané. Na vysvětlenou pouze stručně uveďme: Otázky po správných formách usuzování si jistě v té či oné podobě kladli lidé velmi dávno. Za zakladatele logiky je považován Aristotelés (384-322 př. n. l.). Od té doby byly otázky logiky předmětem zájmu mnoha učenců, bylo nastoleno mnoho otázek a témat. Většina z nich má povahu fundamentálních otázek po podstatě lidského usuzování a je tedy svou povahou filozofická. Teprve koncem 19. a začátkem 20. století došlo k výrazné matematizaci logiky. Při této matematizaci šlo především o formalizaci usuzování v matematice a v tzv. exaktních vědách, ne lidského usuzování obecně. Matematickými prostředky tak byly studovány pouze vybrané aspekty usuzování a vznikl dojem, že ostatní aspekty usuzování (např. vliv času ve tvrzeních obsahujících odkaz na čas jako "Někdy v minulosti pršelo", tvrzení obsahující modality jako "je možné, že prší") se matematickými prostředky nedají studovat. Studium těchto "ostatních aspektů" bylo přisuzováno tzv. filozofické logice. Postupně se však docházelo k tomu, že matematické metody byly používány i ke studiu zmíněných "ostatních aspektů", a ukázalo se, že dělení logiky na matematickou filosofickou je věcně neopodstatněné.

Logika: klasická a neklasická? Klasickou logikou se rozumí logika, ve které předpokládáme, že tvrzení mohou nabývat dvou pravdivostních hodnot (pravda a nepravda), ve které tvrzení mohou být spojována ve tvrzení složitější spojkami "není pravda, že ...", "... a...", "... nebo...", "jestliže ..., pak...", "..., právě když ..." a kvantifikátory "pro každé x ..." a "existuje x, pro které ..." a ve které pravdivostní hodnoty složených tvrzení závisí na pravdivostních hodnotách skládaných tvrzení způsobem, který zná čtenář nejspíš již ze střední školy (viz dále). Jiná logika se považuje za neklasickou (tvrzení mohou mít více hodnot nebo je možné používat i jiné spojky nebo mají spojky jiný význam apod.).

Logika: čistá a aplikovaná? Aplikovanou logikou se někdy rozumí studium problémů, které vznikají při pokusu použít logiku na tu či onu oblast, která má oproti situaci, kdy nás zajímá usuzování vůbec, svá specifika. Tato specifika umožňují řadu dodatečných (zjednodušujících) předpokladů, díky kterým se ve specifických oblastech metodami logiky dosahuje pozoruhodných výsledků. Čistou logikou se v této souvislosti rozumí logika všeobecná, zabývající se formami usuzování bez ohledu na konkrétní oblasti použití. Je zřejmé, že i toto dělení je spíše orientační, vyjadřující víceméně motivace a cíle.

Jaký je vztah logiky a informatiky? Stručně řečeno, bohatý a různorodý. DOPLNIT

- Automatické dokazování.
- Logické programování.
- Logika programů.

- Přibližné usuzování.
- Analýza dat.

Logika: výroková a predikátová? V našem pojednání se budeme nejdříve zabývat tzv. výrokovou logikou, poté tzv. predikátovou logikou. Aby bylo jasno, řekněme už teď, že nejde "o dvě různé logiky". Predikátová logika je rozšířením výrokové logiky (výstižněji řečeno zjemněním výrokové logiky). Bylo by tedy možné výrokovou logiku vynechat a zabývat se rovnou predikátovou logikou. Protože je však výroková logika jednodušší a protože nám navíc umožní ilustrovat základní problémy logiky v dostatečné míře, začneme z didaktických důvodů logikou výrokovou. O tom, jaký je přesně vztah výrokové a predikátové logiky, se zmíníme později.

2 Výroková logika

[1]

2.1 ... k výrokové logice

Výrokem intuitivně chápeme výpověď (tvrzení), u které má smysl uvažovat o její pravdivosti. Tedy např. "Prší." je výrok, "Byl jsem v obchodě a koupil jsem si knihu." je výrok, "2+2=6" je výrok, ale "kniha v obchodě" není výrok, "2+2" není výrok. Výroková logika bývá nazývána logikou výrokových spojek; přitom spojkami se myslí spojky jako "a", "nebo", "jestliže, ... pak ...", "ne" (tj. "není pravda, že ...") apod.[2]

Některé výroky jsou pravdivé (např. "2+2=4"), některé jsou nepravdivé (např. "2+2=6"). O tvrzení, které je pravdivé, řekneme, že má pravdivostní hodnotu 1 (pravda); o tvrzení, které je nepravdivé, řekneme, že má pravdivostní hodnotu 0 (nepravda).

U některých tvrzení má smysl uvažovat o jejich pravdivosti (jsou tedy výroky), např. "Člověk s výškou 180cm je vysoký", přesto se však zdráháme říci, že dané tvrzení je pravdivé nebo, že není pravdivé (je nepravdivé). Pravdivostní hodnota, kterou bychom mu přiřadili, by byla někde mezi 0 a 1, např. řekneme-li, že "Člověk vysoký 180cm je vysoký" má pravdivostní hodnotu 0.8, říkáme tím, že je pravdivé ve stupni 0.8, tj. že je skoro pravdivé. V následujícím se omezíme na (tvrzení, které mohou mít jen) dvě pravdivostní hodnoty (0 a 1). Poznamenejme pouze, že studiem více pravdivostních hodnot se zabývá tzv. fuzzy logika (ta je v současné době intenzívně zkoumána).

U některých tvrzení závisí pravdivostní hodnota na čase, např. "Za týden bude pršet." Takovými tvrzeními se zabývá tzv. logika času (temporální logika).

[1] Též výrokový počet nebo výrokový kalkul.

[2] Všimněme si, že pojem spojka zde používáme v širším významu než bývá běžné: spojkou chápeme jazykový výraz, jehož použitím na výroky dostaneme nový výrok. Např. výrok "Jestliže prší, pak je mokro." vznikl použitím spojky "jestliže..., pak ..." na výroky "prší" a "je mokro" apod.

Ta má pro informatiku zásadní význam. My se s ní však nesetkáme (resp. setkáme se jen okrajově) a budeme faktor času ignorovat, tj. budeme se zabývat tvrzeními, jejichž pravdivost na čase závislá není.

Posledním omezení (tj. zjednodušení), které přijmeme, se týká výrokových spojek. Podle výše uvedeného, je spojkou nejen "ne" (tj. "není pravda, že ..." např. v tvrzení "Není pravda, že prší"), ale také "nutně" (tj. "nutně platí, že ..." např. v tvrzení "Nutně platí, že prší"), "možná" (tj. "je možné, že ..." např. v tvrzení "Je možné, že prší"), "ví se" (tj. "ví se, že ..." např. v tvrzení "Ví se, že prší") apod. Takové spojky je možné zkoumat (spojky "nutně" a "možná" zkoumá tzv. modální logika, spojky "ví se" apod. zkoumá tzv. epistemická logika), my to však dělat nebudeme a omezíme se jen na tzv. spojky klasické.

Poznámka 2.1 Všimněme si, že už teď jsme učinili řadu zjednodušujících předpokladů.

Všimněme si, že v přirozeném jazyce existují tvrzení, která znamenají to samé, ale jsou různá: např. dvojice tvrzení "Neprší." a "Není pravda, že prší" nebo trojice tvrzení "Možná prší.", "Je možné, že prší." a "Možná je pravda, že prší."

Jak jsme zmínili výše, úkolem logiky je zkoumat usuzování a související problémy, avšak bez ohledu na obsah toho, co je (o čem je) usuzováno. Odhlédneme-li u usuzování od obsahu, zbyde forma (např. u výše uvedeného odvození tvrzení "silnice jsou mokré" ze dvou tvrzení "prší" a "jestliže prší, (pak) jsou silnice mokré" zbyde po odhlédnutí od obsahu (abstrakce od obsahu) forma tohoto odvození, kterou je možné zachytit pomocí: z "A" a "$A \Rightarrow B$" lze odvodit "B"). V tomto smyslu jde tedy v logice o formu usuzování.

Chceme se tedy soustředit usuzování o výrocích, přitom nás nezajímá obsah, ale pouze forma usuzovaného. Na jedné straně tím v jistém smyslu ztratíme, neboť naše rozlišovací schopnost odhlédnutím od obsahu klesne (výroky, popř. úsudky se stejnou formou, ale různým obsahem, pro nás budou nerozlišitelné). Na druhé straně, soustředěním se pouze na formu nám umožní objevit zákonitosti, kterými se řídí usuzování nad jakýmikoli výroky, tedy výroky s libovolným obsahem. Výsledky, které získáme, budeme tedy moci použít na libovolné výroky. Další výhodou odlhlédnutí od obsahu je bezesporu fakt, že zbavíme-li se nepodstatného (v našem případě obsahu), budeme lépe vidět podstatné (v našem případě zákonitosti usuzování).

Je důležité upozornit na to, že nám jde o dvě věci: Za prvé, chceme studovat formy usuzování a související aspekty. Za druhé, chceme naše úvahy zpřesnit tak, abychom ke studiu forem usuzování mohli použít matematické metody. Čtenář si jistě uvědomil, že zatím jsme se pohybovali na intuitivní (a tedy nepřesné, matematickými metodami neuchopitelné) úrovni. Tak například, řekli jsme, že výrok je výpověď, u které má smysl uvažovat o její pravdivosti. Nedefinovali jsme přesně, co rozumíme spojením "výpověď, u které má smysl uvažovat o její pravdivosti" a nechali jsme toto na čtenářově intuitivním porozumění. V dalším přistoupíme k vybudování formálního systému výrokové logiky.

2.2 Základní syntaktické pojmy

Chceme-li zkoumat formy usuzování o výrocích bez ohledu na jejich obsah, bude užitečné označovat výroky pomocí symbolů (to jsme ostatně již učinili, když jsme použili A a B pro označení výroků při ukázce odhlédnutí od obsahu). Některé výroky vznikly z jiných pomocí spojek, jiné spojky neobsahují (např. "Prší."), a jsou tedy v tomto smyslu nedělitelné (atomické). Atomické výroky budeme označovat speciálními symboly, kterým budeme říkat *výrokové symboly*. Spojky, kterými se výroky spojují ve složené výroky, budeme označovat *symboly výrokových spojek*. Tak budeme moci zapisovat formu výroků: např. označují-li symboly p a q po řadě výroky "prší" a "silnice jsou mokré" a označuje-li $\Rightarrow$ spojku "jestliže ..., pak ...", výrok "jestliže prší, pak silnice jsou mokré" je zapsán výrazem $p \Rightarrow q$. Navíc, zapomeneme-li na obsahy výroků označených pomocí p a q, reprezentuje výraz $p \Rightarrow q$ formu výroku. Abychom mohli pohodlně a přehledně zapisovat složené výroky, budeme používat pomocné symboly (většinou závorky různých druhů). Dostali jsme se k tomu, co se nazývá jazyk výrokové logiky:

Definice 2.2 ***Jazyk výrokové logiky*** se skládá z

- ***výrokových symbolů*** $p, q, r, \ldots$, popř. s indexy, p_1, p_2; předpokládáme, že máme neomezeně mnoho (spočetně mnoho) výrokových symbolů;
- ***symbolů výrokových spojek*** $\neg$ (negace), $\Rightarrow$(implikace), popř. dále $\wedge$ (konjunkce), $\vee$ (disjunkce), $\Leftrightarrow$ (ekvivalence);
- ***pomocných symbolů*** (,), [,], atd. (různé druhy závorek).

Než přistoupíme k další definici, musíme objasnit tu část definice jazyka výrokové logiky, která se týká spojek (ta je v Definici 2.2 nejednoznačná). V zásadě jde o to, že některé spojky je možné nahradit jinými. Tak například výrok "Prší a je zima." je pravdivý, právě když je pravdivý výrok "Není pravda, že neprší nebo není zima." To platí pro jakékoli výroky. Výrok tvaru "A a B" má stejnou pravdivostní hodnotu jako výrok "Není pravda, že neplatí A nebo neplatí B". Je tedy vidět, že spojku "a" (použitou v prvním výroku) lze nahradit pomocí spojek "ne" a "nebo" (použitých ve druhém výroku). Můžeme tedy postupovat tak, že některé spojky zvolíme za základní a další budeme považovat za odvozené (to zatím vidíme intuitivně, za okamžik uvidíme, jak to provést korektně). To má své výhody, které plně doceníme až v následujícím textu (výhody jsou především technické: budeme mít kratší definice a kratší důkazy).

Shrňme tedy: Jde nám o spojky "ne", "a", "nebo", "jestliže ..., pak ...", "..., právě když ...". Pracovat s nimi můžeme dvojím způsobem. Za prvé, můžeme je (resp. jim odpovídající symboly v jazyce) všechny považovat za základní. Za druhé, můžeme vybrat jen jistou skupinu spojek, která stačí pro vyjádření ostatních spojek, a tuto skupinu považovat za základní; ostatní spojky pak můžeme považovat za odvozené ze základních (v tomto případě stačí mít v jazyce jen symboly základních spojek; symboly ostatních spojek můžeme používat, ale chápeme je jako definované pomocí symbolů pro základní spojky—to za chvíli uvidíme). První postup je přirozenější (neboť žádnou spojku neupřednostňujeme), druhý je technicky výhodnější (neboť pracujeme s menším množstvím spojek). My zvolíme druhý postup. Je důležité zmínit, že oba postupy jsou ekvivalentní (to my však neuvidíme, protože první způsob ignorujeme; zvídavý čtenář může první způsob domyslet, popř. najít v literatuře; podotkněme pouze, že říkáme-li, že oba postupy jsou ekvivalentní, myslíme tím, že všechna tvrzení, ke kterým dospějeme naším způsobem, jsou (v odpovídající podobě) platná i při použití prvního způsobu; DODELAT, nejspíš do cvičení). Správně by tedy v definici symbolů pro výrokové spojky mělo být při našem přístupu uvedeno

- *symbolů výrokových spojek* $\neg$ (negace), $\Rightarrow$(implikace);

Uvedení symbolů dalších spojek má za úkol stručně vyjádřit právě vysvětlenou situaci.

Přistupme k další definici.

Definice 2.3 Nechť je dán jazyk výrokové logiky. ***Formule*** daného jazyka výrokové logiky je definována následovně:

- každý výrokový symbol je formule (tzv. atomická formule);
- jsou-li φ a ψ formule, jsou i $\neg\varphi$ a $(\varphi \Rightarrow \psi)$ formule.

Příklad 2.4 Formulemi jsou tedy jisté konečné posloupnosti symbolů jazyka výrokové logiky. Např. posloupnosti p, q_1, $\neg p$, $(p \Rightarrow q)$, $(\neg p \Rightarrow (q \Rightarrow \neg r))$ jsou formule, naproti tomu posloupnosti pp, $((p\neg$, $\Rightarrow p$ nejsou formule. Posloupnost $(\neg p \Rightarrow (q \Rightarrow \neg r))$ je formule, protože: r je formule (atomická), tedy i $\neg r$ je formule, což spolu s tím, že q je formule, dává, že $(q \Rightarrow \neg r)$ je formule; dále je p a tedy i $\neg p$ formule, a tedy konečně i $(\neg p \Rightarrow (q \Rightarrow \neg r))$ je formule.

Zavedeme nyní symboly $\wedge$, $\vee$, $\Leftrightarrow$, jak jsme ohlásili výše. Jsou-li, φ a ψ posloupnosti symbolů, pak posloupnosti $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \Leftrightarrow \psi)$ jsou zkratky za následující posloupnosti:

$$\begin{aligned}
(\varphi \wedge \psi) \quad & \text{je zkratkou za } \neg(\varphi \Rightarrow \neg\psi),\\
(\varphi \vee \psi) \quad & \text{je zkratkou za } (\neg\varphi \Rightarrow \psi),\\
(\varphi \Leftrightarrow \psi) \quad & \text{je zkratkou za } ((\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi).
\end{aligned}$$

Vezměme např. posloupnost $((p \wedge q) \wedge r)$. Ta je zkratkou za posloupnost $\neg((p \wedge q) \Rightarrow \neg r)$. Protože je dále $(p \wedge q)$ zkratkou za $\neg(p \Rightarrow \neg q)$, je původní posloupnost $((p \wedge q) \wedge q)$ zkratkou za posloupnost $\neg(\neg(p \Rightarrow \neg q) \Rightarrow \neg r)$, která jej již formulí. Podobně je $(p \vee (q \vee r))$ zkratkou za formuli $(\neg p \Rightarrow (\neg q \Rightarrow r))$ a $((p \wedge q) \vee r)$ je zkratkou za $(\neg(\neg(p \Rightarrow \neg q)) \Rightarrow r)$. Uvědomme si dále, že protože $(\varphi \Leftrightarrow \psi)$ je zkratkou za $((\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi))$, je také zkratkou za $\neg((\varphi \Rightarrow \psi) \Rightarrow \neg(\psi \Rightarrow \varphi))$. Jak je z právě uvedeného zřejmé, chápeme vztah "být zkratkou za" mezi posloupnostmi znaků za tranzitivní: je-li $\mathcal{A}$ zkratkou za $\mathcal{B}$ a $\mathcal{B}$ zkratkou za $\mathcal{C}$, je $\mathcal{A}$ zkratkou za $\mathcal{C}$. Některé posloupnosti sestavené ze symbolů jazyka výrokové logiky a symbolů $\wedge$, $\vee$ a $\Leftrightarrow$ jsou zkratkami za posloupnosti, které již neobsahují $\wedge$, $\vee$, ani $\Leftrightarrow$; některé posloupnosti jsou zkratkou za posloupnosti, které jsou formulemi. Lze lehce ukázat, že posloupnost $\mathcal{A}$ je zkratkou za formuli výrokové logiky (tj. obsahuje jen spojky $\neg$ a $\Rightarrow$), právě když je formulí výrokové logiky podle definice, která připouští i $\wedge$, $\vee$ a $\Leftrightarrow$ jako symboly spojek (DO CVIČENÍ). Posloupnosti, které jsou zkratkami formulí, nejsou samy o sobě formule, pro jednoduchost jim však formule říkat budeme. Tedy řekneme například "formule $(p \wedge \neg q)$", přestože bychom měli správně říct "formule, jejíž zkratkou je $(p \wedge \neg q)$".

Poznámka 2.5 (1) Formule výrokové logiky tedy odpovídají výrokům, reprezentují formu výroků.

(2) Všimněme si, že správně bychom měli říkat "formule daného jazyka $\mathcal{J}$ výrokové logiky". My však v případě, že jazyk $\mathcal{J}$ je zřejmý z kontextu, popř. není důležitý, budeme říkat pouze "formule výrokové logiky" nebo jen "formule".

Poznámka 2.6 (konvence o přehlednějším zápisu formulí) Jak zná čtenář z aritmetiky, je pro zjednodušení zápisu a čtení užitečné vynechávat závorky tam, kde neutrpí jednoznačnost čtení formule (např. místo $(p \Rightarrow q)$ můžeme klidně psát jen $p \Rightarrow q$). Dále se dohodněme na prioritách symbolů spojek a symbolů $\wedge$, $\vee$ a $\Leftrightarrow$: od největší po nejmenší je to $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$. To nám umožní vynechávat závorky. Tak např. místo $(p \wedge (q \wedge r))$ můžeme psát jen $p \wedge (q \wedge r)$, místo $(p \Rightarrow ((p \wedge q) \vee r))$ jen $p \Rightarrow p \wedge q \vee r$ apod. (čtenář jistě konvence o vynechávání závorek zná, proto je nebudeme rozvádět).

Poznámka 2.7 Formule jsou tedy jisté řetězce nad abecedou (označme ji Σ) sestávající z výrokových symbolů, symbolů spojek (tj. $\neg$ a $\Rightarrow$) a pomocných symbolů (tj. závorek; uvažujme nyní pouze "kulaté" závorky "(" a ")"). Množina Fml všech formulí (daného jazyka výrokové logiky)je tedy jazykem nad abecedou Σ. Tento jazyk je možné chápat jako jistou algebru, a to následovně. Definujme na množině Σ^* všech řetězců nad Σ unární operaci $f^{\neg}$ a binární operaci $f^{\Rightarrow}$ předpisem

$$f^{\neg}(u) = \neg\, u$$
$$f^{\Rightarrow}(u, v) = (u \Rightarrow v)$$

pro každé $u, v \in \Sigma^*$. Je tedy např. $f^{\neg}(p_1 p_2 \neg\Rightarrow q \Rightarrow) = \neg\, p_1 p_2 \neg\Rightarrow q \Rightarrow$, $f^{\neg}(p \Rightarrow q) = \neg\,(p \Rightarrow q)$, $f^{\Rightarrow}(\neg, p \neg (q \Rightarrow) = (\neg\Rightarrow p \neg (q \Rightarrow)$, $f^{\Rightarrow}((p \Rightarrow p), \neg q) = ((p \Rightarrow p) \Rightarrow \neg q)$. Je-li P množina všech výrokových symbolů, pak Fml je právě nosičem podalgebry algebry $\langle \Sigma^*, f^{\neg}, f^{\Rightarrow} \rangle$ generované množinou P. Skutečně, ... (DODELAT, je možné použít princip strukturální indukce)

Formule byly definovány tzv. induktivním (nebo rekurzívním) způsobem. Kromě toho, že takový způsob nám umožnil konečným způsobem definovat nekonečnou množinu (množina formulí je nekonečná), umožňuje nám navíc elegantně dokazovat tvrzení tvaru "Každá formule má vlastnost $\mathcal{V}$". Lehce lze totiž nahlédnout, že platí následující tvrzení.

Věta 2.8 (důkaz strukturální indukcí pro formule) *Nechť $\mathcal{V}$ je vlastnost formulí. Nechť platí, že*

- *každý výrokový symbol má vlastnost $\mathcal{V}$;*
- *mají-li formule φ a ψ vlastnost $\mathcal{V}$, pak vlastnost $\mathcal{V}$ mají i formule $\neg\,\varphi$ a $(\varphi \Rightarrow \psi)$.*

Pak vlastnost $\mathcal{V}$ má každá formule.

Důkaz. Použijeme princip strukturální indukce z Věty 7.1. Nechť $\mathbf{A}$, $\mathbf{B}$ a C (viz Větu 7.1) jsou po řadě algebry $\langle \Sigma^*, f^{\neg}, f^{\Rightarrow}\rangle$, $\langle \mathrm{Fml}, f^{\neg}, f^{\Rightarrow}\rangle$ a množina P všech výrokových symbolů (viz Poznámku 2.7). Dokazované tvrzení je přímým důsledkem Věty 7.1 a faktu, že $\langle \mathrm{Fml}, f^{\neg}, f^{\Rightarrow}\rangle$ je podalgebra generovaná množinou P. □

(K UVAZE: zavest spec. znaceni pro mnozinu symbolu jazyka)

Příklad 2.9 Ukažme si použití důkazu strukturální indukcí na jednoduchém příkladě. Chceme dokázat, že počet levých závorek je v každé formuli roven počtu pravých závorek. To tvrzení je téměř zřejmé. Nic to však nemění na faktu, že je třeba ho dokázat; další tvrzení týkající se formulí již nebudou tak zřejmá. Důkaz strukturální indukcí použijeme následovně: Vlastnost V, o kterou zde jde, je "mít stejný počet levých a pravých závorek". To, že každý výrokový symbol má vlastnost V je zřejmé, neboť výrokový symbol nemá žádné závorky (tedy má 0 levých a 0 pravých). Mají-li formule φ a ψ vlastnost V, tj. mají-li stejný počet levých i pravých závorek, pak vlastnost V mají i formule $\neg\,\varphi$ a $(\varphi \Rightarrow \psi)$. Opravdu, označíme-li pomocí $l(A)$ a $r(A)$ počet levých a pravých závorek v posloupnosti A, pak pomocí indukčního předpokladu (tím je tvrzení, že $l(\varphi) = r(\varphi)$ a $l(\psi) = r(\psi)$) dostáváme $l(\neg\,\varphi) = l(\varphi) = r(\varphi) = r(\neg\,\varphi)$ a podobně $l(\varphi \Rightarrow \psi) = l(\varphi) + l(\psi) + 1 = r(\varphi) + r(\psi) + 1 = r(\varphi \Rightarrow \psi)$.

Příklad 2.10 (cvičení) Dokažte strukturální indukcí, že nahradíme-li ve formuli výrokové symboly formulemi, dostaneme opět formuli (přitom jeden výrokový symbol můžeme na různých místech nahradit různými formulemi). Stejné tvrzení platí, nahrazujeme-li podformule dané formule formulemi (přitom podformulí dané formule je formule, která je nachází v dané formuli jako podřetězec; tj. např. formule $(p \Rightarrow q) \Rightarrow p$ má podformule sebe samu, $(p \Rightarrow q)$, p a q).

2.3 Základní sémantické pojmy

Zatím jsme se věnovali jen tzv. syntaktické stránce výrokové logiky. Víme, co je to jazyk výrokové logiky, co jsou to formule. Zatím však nevíme, co to je pravdivá formule apod. Formule jsou jisté posloupnosti symbolů jazyka, samy o sobě však nemají žádný význam. Přiřazení významu syntaktickým objektům je záležitostí tzv. sémantiky. Právě sémantikou výrokové logiky se v dalším budeme věnovat.

Definice 2.11 ***(Pravdivostní) ohodnocení*** je libovolné zobrazení výrokových symbolů daného jazyka výrokové logiky do množiny $\{0, 1\}$, tj. libovolné zobrazení e přiřazující každému výrokovému symbolu p hodnotu $e(p) \in \{0, 1\}$.

Poznámka 2.12 (1) 0 a 1 reprezentují pravdivostní hodnoty nepravda a pravda.

(2) Význam ohodnocení e můžeme chápat následovně: Výrokové symboly označují výroky. Zadáme-li ohodnocení e, pak výrokový symbol p označuje výrok, který má pravdivostní hodnotu $e(p)$. Na to lze nahlížet dvěma způsoby. Za prvé, výrok označený symbolem p známe a považujeme jej za pravdivý (nepravdivý). Pak položíme $e(p) = 1$ ($e(p) = 0$). Za druhé, pravdivostní hodnotu výroku označeného symbolem p neznáme. Položíme-li pak $e(p) = 1$ ($e(p) = 0$), omezujeme se na situace, kdy výrok označený symbolem p je pravdivý (např. když nás zajímá, co vyplývá z pravdivosti zmiňovaného výroku apod.).

Je-li dáno ohodnocení e, můžeme říci, co je to pravdivostní hodnota formule. Pravdivostní hodnota libovolné formule je pravdivostním ohodnocením jednoznačně určena a je definována následovně.

Definice 2.13 Nechť je dáno ohodnocení e. Pravdivostní hodnota formule φ při ohodnocení e, označujeme ji $\|\varphi\|_e$, je definována následovně:

- $\|p\|_e = e(p)$ pro výrokový symbol p;
- $\| \neg \varphi\|_e = 1$ pokud $\|\varphi\|_e = 0$, $\| \neg \varphi\|_e = 0$ pokud $\|\varphi\|_e = 1$; a $\|(\varphi \Rightarrow \psi)\|_e = 1$, pokud $\|\varphi\|_e = 0$ nebo $\|\psi\|_e = 1$, $\|(\varphi \Rightarrow \psi)\|_e = 0$ jinak.

Je-li $\|\varphi\|_e = 1$ ($\|\varphi\|_e = 0$), říkáme, že formule φ je při ohodnocení e pravdivá (nepravdivá). Uvědomme si, že nemá smysl říci "formule φ je pravdivá" nebo "nepravdivá" (musíme říci, při jakém ohodnocení!).

Poznámka 2.14 Lehce je vidět (ověřte si), že $\|(\varphi \wedge \psi)\|_e = 1$, právě když $\|\varphi\|_e = 1$ a $\|\psi\|_e = 1$ (a tedy $\|(\varphi \wedge \psi)\|_e = 0$ jinak, tj. pokud $\|\varphi\|_e = 0$ nebo $\|\psi\|_e = 0$); $\|(\varphi \vee \psi)\|_e = 1$, právě když $\|\varphi\|_e = 1$ nebo $\|\psi\|_e = 1$ (a tedy $\|(\varphi \vee \psi)\|_e = 0$ jinak, tj. pokud $\|\varphi\|_e = 0$ i $\|\psi\|_e = 0$); $\|(\varphi \wedge \psi)\|_e = 1$, právě když $\|\varphi\|_e = \|\psi\|_e$.

ALTERNATIVNĚ: zadávání pravdivostních funkcí logických spojek tabulkou:

a	$\neg a$
0	1
1	0

$\rightarrow$	0	1
0	1	1
1	1	0

Definice 2.15 Formule se nazývá ***tautologie*** (***kontradikce***), je-li při každém ohodnocení pravdivá (nepravdivá). Formule se nazývá ***splnitelná*** , je-li při nějakém ohodnocení pravdivá.

Je-li φ tautologie, píšeme také $\models \varphi$, popř. $\|\varphi\| = 1$. Zjistit pravdivostní hodnotu formule při daném ohodnocení je snadné, lze to provést přímo z definice. Zjistit, zda je formule tautologií, kontradikcí, popř. splnitelná, však dle definice znamená zjistit její pravdivostní hodnotu (v nejhorším případě) pro všechna ohodnocení. Díky následujícímu tvrzení to však není nutné.

Je-li φ formule výrokové logiky, píšeme také $\varphi(p_1, \ldots, p_n)$, chceme-li zdůraznit, že všechny výrokové symboly vyskytující se v φ jsou mezi $p_1, \ldots, p_n$ (tj. žádný jiný výrokový symbol než některý z $p_1, \ldots, p_n$ se v φ nevyskytuje).

Lemma 2.16 *Platí-li pro ohodnocení e a e', že $e(p_1) = e'(p_1)$, ..., $e(p_n) = e'(p_n)$, pro každou formuli $\varphi(p_1, \dots, p_n)$ platí $\|\varphi\|_e = \|\varphi\|_{e'}$.*

Příklad 2.17 Dokažte předchozí lemma strukturální indukcí (lemma je téměř zřejmé, procvičíte si však strukturální indukci).

Jinak řečeno, pravdivostní hodnota formule závisí jen na tom, jaké hodnoty přiřazuje dané ohodnocení výrokovým symbolům, které se ve formuli vyskytují. Pro n výrokových symbolů $p_1, \dots, p_n$ existuje právě 2^n různých ohodnocení symbolů $p_1, \dots, p_n$ (každému p_i se přiřazuje 0 nebo 1). Tyto úvahy jsou základem tzv. tabulkové metody pro zjištění pravdivostních hodnot formule: Tabulka pro formuli, která obsahuje n výrokových symbolů, má 2^n řádků, každý řádek odpovídá pravdivostnímu ohodnocení, na konec každého řádku (tj. do posledního sloupce) zapíšeme pravdivostní hodnotu formule při odpovídajícím ohodnocení (viz přednášky). Formule je tautologií (kontradikcí, splnitelnou), právě když jí odpovídající tabulka pravdivostních hodnot má v posledním sloupci samé 1 (samé 0, aspoň jednu 1).

Příklad 2.18 Např. $p \vee \neg\, p$ je tautologie, $p \wedge \neg\, p$ je kontradikce, $p \vee q$ je splnitelná (která není tautologií).

Definice 2.19 Formule ψ ***sémanticky plyne*** z formule φ (značíme $\varphi \models \psi$), jestliže ψ je pravdivá při každém ohodnocení, při kterém je pravdivá φ. Pokud ψ sémanticky plyne z φ a naopak, říkáme, že φ a ψ jsou sémanticky ekvivalentní. Obecněji, ψ sémanticky plyne z množiny T formulí (značíme $T \models \psi$), je-li ψ pravdivá při každém ohodnocení, při kterém je pravdivá každá formule z T.

Poznámka 2.20 Právě zavedené použití symbolu $\models$ je v souladu s dříve zavedeným $\models \varphi$ pro označení faktu, že φ je tautologie. Skutečně, chápeme-li $\models$ ve smyslu sémantického vyplývání a chápeme-li dále $\models \varphi$ jako $\emptyset \models \varphi$, pak $\models \varphi$ znamená, že pro každé ohodnocení e platí, že je-li při e pravdivá každá formule z $\emptyset$ (žádná taková ale není), pak je při e pravdivá také φ. To však znamená, že pro každé ohodnocení e je formule φ pravdivá.

Ke každé formuli výrokové logiky existuje s ní (sémanticky) ekvivalentní formule, která je ve tvaru úplné konjunktivní normální formy (úplné disjunktivní normální formy); viz přednášky (literál je výrokový symbol nebo jeho negace; úplná elementární konjunkce (disjunkce) na danou množinou V výrokových symbolů je konjunkce (disjunkce) literálů, ve které se každý výrokový symbol z V vyskytuje právě v jednom literálu; úplná disjunktivní (konjunktivní) normální forma (nad V) je disjunkce (konjunkce) úplných elementárních konjunkcí (disjunkcí) (nad V); konstrukce ÚDNF dané formule φ: z tabulky pravdivostních hodnot φ se vyberou řádky, pro které je formule pravdivá a pro každý takový řádek se vytvoří úplná elementární konjunkce takto: pro každý výrokový symbol p formule φ se vytvoří odpovídající literál, přitom má-li p v ohodnocení příslušném tomuto řádku hodnotu 1, je tím literálem přímo p, má-li hodnotu 0, je tím literálem $\neg\, p$; takto vybrané literály se spojí konjunkcí; takto vytvořené konjunkce se spojí disjunkcí — zdůvodněte, proč takto skutečné vznikne formule ekvivalentní původní formuli φ; duálně se vytvoří úplná disjunktivní normální forma).

Příklad 2.21 Zjistěte, zda formule ψ sémanticky plyne z φ: (a) φ je $(p \Cap q) \Cup r$, ψ je $p \Rightarrow (q \Cup r)$; (b) φ je $(p \Cap q) \Cup r$, ψ je $p \Rightarrow (q \Cup \neg r)$. Řešení: V prvním případě ano, ve druhém ne.

Příklad 2.22 (1) Přesvědčte se, že je-li $\psi \models \varphi$ a $\varphi \models \chi$, pak $\psi \models \chi$. [řešení: jednoduchou úvahou: máme ukázat $\psi \models \chi$; nechť e je ohodnocení při kterém je ψ pravdivá; dle předpokladu $\psi \models \varphi$ je při e pravdivá také φ, a tedy dle předpokladu $\varphi \models \chi$ je při e pravdivá také χ, což jsme měli ukázat].

(2) Dokažte, že pro libovolnou množinu T formulí a formule φ, ψ platí

$$T, \varphi \models \psi \text{ právě když } T \models \varphi \Rightarrow \psi.$$

To je sémantická podoba tzv. věty o dedukci (viz později). Přitom T, φ znamená $T \cup \{\varphi\}$ (tato dohoda se v logice používá často). [Důkaz je velmi snadný, stačí si rozmyslet, co máme dokázat: řešení: Předpokládejme $T, \varphi \models \psi$ a dokažme $T \models \varphi \Rightarrow \psi$. Máme tedy dokázat, že je-li e ohodnocení, při kterém jsou pravdivé všechny formule z T, je při e pravdivá i formule $\varphi \Rightarrow \psi$. Kdyby ale při e nebyla pravdivá formule $\varphi \Rightarrow \psi$, musela by být při e φ pravdivá a ψ nepravdivá (z definice pravdivostních hodnot implikace). Je-li ale při e pravdivá φ i všechny formule z T, pak je dle předpokladu $T, \varphi \models \psi$ pravdivá i ψ, což je spor s tím, že ψ je nepravdivá. Naopak, předpokládejme $T \models \varphi \Rightarrow \psi$ a dokažme $T, \varphi \models \psi$. Máme dokázat, že je-li e ohodnocení, při kterém je pravdivá φ i všechny formule z T, je při něm pravdivá i ψ. Dle předpokladu je ovšem při e pravdivá i $\varphi \Rightarrow \psi$ a protože je při e pravdivá i φ, je při e pravdivá i ψ, což jsme měli dokázat.]

2.4 Axiomatický systém výrokové logiky

Idea axiomatického systému: V každodenním životě se dobíráme dalších věcí následovně: Vyjdeme z toho, co víme nebo co předpokládáme a použitím usuzování (i v běžném životě říkáme "logicky správného usuzování") odvozujeme z předpokladů nové a nové závěry. Snaha přijít na kloub tomu, jak vlastně člověk usuzuje, provází člověka po staletí. Uvědomme si, že zejména z pohledu informatiky je otázka, jak člověk usuzuje, zcela zásadní. Představa, že do počítače "nasypeme" to, co víme nebo předpokládáme (o nějaké oblasti, která nás zajímá), a že počítač nám pomocí algoritmizovaného (automatizovaného) postupu usuzování nabídne závěry, je totiž velmi lákavá (i když se o tom v tomto textu nedozvíme, podotkněme, že tato problematika patří mezi podrobně studované otázky teoretické informatiky, které mají aplikaci v umělé inteligenci, tzv. logickém programování; jak jistě čtenář uzná, tyto otázky mají svůj půvab a zcela zásadní charakter i z hlediska filozofického (mohou počítače myslet?)).

Otázka po podstatě usuzování, které v běžném životě nazýváme logicky správným, je základní otázkou matematické logiky. V logice postupujeme při formalizaci výše uvedeného následovně:

To, co víme nebo předpokládáme, se nazývá axiomy, popř. předpoklady (předpokládáme vždy výroky, tj. tvrzení, a ty jsou v logice reprezentovány formulemi, axiomy jsou tedy formule). Elementární úsudkový krok je realizován pomocí tzv. odvozovacího (inferenčního) pravidla (to je tvaru "z formulí tvaru toho a toho odvoď formuli tu a tu"). Záznam odvozování, provedený tak, že za sebe napíšeme tvrzení (formule), ke kterým se postupně dobíráme tak, že začneme předpoklady (axiomy) a pokračujeme tvrzeními (formulemi), které z předchozích tvrzení (formulí) plynou pomocí elemenntárních úsudkových kroků (odvozovacích pravidel), nazýváme důkaz. Tvrzení považujeme za zdůvodnitelné (dokazatelné) z dané množiny předpokladů (axiomů, popř. dalších dodatečných předpokladů), pokud k němu existuje důkaz. Takto navržený formální systém potom nazýváme axiomatickým systémem.

V následujícím uvedeme jeden konkrétní axiomatický systém výrokové logiky a prozkoumáme jeho vlastnosti. Axiomatický systém má axiomy a odvozovací pravidla.

Náš axiomatický systém má za axiomy jakékoli formule, které jsou některého z následujících tvarů (tzv. axiomových schémat):

$$\varphi \Rightarrow (\psi \Rightarrow \varphi) \tag{1}$$
$$(\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \chi)) \tag{2}$$
$$(\neg\, \psi \Rightarrow \neg\, \varphi) \Rightarrow (\varphi \Rightarrow \psi). \tag{3}$$

Náš axiomatický systém má dále jedno odvozovací pravidlo nazývané *modus ponens* (MP, pravidlo odloučení), které říká

z formulí φ a $\varphi \Rightarrow \psi$ odvoď ψ.

Řekneme, že formule je axiomem, jestliže vznikne z některého z axiomových schémat (1)–(3) nahrazením symbolů φ, ψ, χ po řadě nějakými formulemi $\theta_1, \theta_2, \theta_3$. Tedy např. $\neg\, (p \Rightarrow p) \Rightarrow (\neg\neg\, p \Rightarrow \neg\, (p \Rightarrow p))$ je axiom, neboť vznikne z (1) nahrazením φ a ψ po řadě formulemi $\neg\, (p \Rightarrow p)$ a $\neg\neg\, p$.

Definice 2.23 ***Důkaz*** formule φ z množiny T formulí je libovolná posloupnost $\varphi_1, \ldots, \varphi_n$, pro kterou platí, že $\varphi_n = \varphi$ a každá φ_i

- je axiomem
- nebo je formulí z T
- nebo plyne z předchozích formulí důkazu pomocí odvozovacího pravidla MP (tj. existují $j, k < i$ $(1 \le j \le n,\ 1 \le k \le n)$ tak že φ_k je formule $\varphi_j \Rightarrow \varphi_i$).

Formule se nazývá ***dokazatelná*** z T, existuje-li důkaz této formule z T (zapisujeme $T \vdash \varphi$, popř. jen $\vdash \varphi$, je-li T prázdná množina).

Poznámka 2.24 (1) Máme tedy přesný pojem důkazu v našem axiomatickém systému (pojem důkazu se tak přenáší z úrovně intuice na přesnou formální úroveň).

(2) Máme dva pojmy vyplývání formule z množiny formulí: sémantické vyplývání ($T \models \varphi$) a syntaktické vyplývání ($T \vdash \varphi$). Jak spolu souvisí, uvidíme později (viz věta o úplnosti).

(3) Speciálně máme dva pojmy platnosti formule: $\models \varphi$ označuje platnost φ v sémantickém smyslu (pravdivost), $\vdash \varphi$ označuje platnost φ v syntaktickém smyslu (dokazatelnost).

Pojem důkaz je matematický objekt (formální pojem), se kterým lze pracovat jako s každým jiným matematickým objektem. Ukažme to na následujícím pozorování.

Pozorování 2.25 *Pro každou množinu T formulí a formule φ, ψ platí, že z $T \vdash \varphi \Rightarrow \psi$ a $T \vdash \varphi$ plyne $T \vdash \psi$.*

Důkaz. Máme tedy dokázat, že jsou-li z T dokazatelné formule $\varphi \Rightarrow \psi$ a φ, pak je z T dokazatelná i formule ψ. Jsou-li však z T dokazatelné formule $\varphi \Rightarrow \psi$ a φ, znamená to, že existuje důkaz (tj. posloupnost formulí splňující podmínky toho, aby byla důkazem) $\chi_1, \ldots, \chi_n$ formule $\varphi \Rightarrow \psi$ z T (tj. χ_n je formulí $\varphi \Rightarrow \psi$) a že existuje důkaz $\theta_1, \ldots, \theta_m$ formule φ z T (tj. θ_m je formulí φ). Nyní však stačí vzít posloupnost $\chi_1, \ldots, \chi_n, \theta_1, \ldots, \theta_m, \psi$—ta je totiž důkazem formule ψ z T. Abychom se o tom přesvědčili, stačí ověřit podmínky z definice pojmu důkaz. Vezměme libovolnou formuli uvažované posloupnosti. Pak je to buď formule χ_i (pro nějaké i) nebo formule θ_j nebo formule ψ. V prvním případě (je to χ_i) uvažujme takto: protože posloupnost $\chi_1, \ldots, \chi_n$ je důkazem, je χ_i axiomem nebo je formulí z T nebo plyne z nějakých předchozích χ_j, χ_k pomocí MP. Ve druhém případě uvažujme podobně. Ve třetím případě plyne uvažovaná formule (je to ψ) pomocí MP z formulí χ_n (což je $\varphi \Rightarrow \psi$) a θ_m (což je φ). Vidíme tedy, že posloupnost $\chi_1, \ldots, \chi_n, \theta_1, \ldots, \theta_m, \psi$ je důkazem ψ z T, tj. $T \vdash \psi$. □

Definice 2.26 Množina T formulí se nazývá ***sporná*** (nekonzistentní), jestliže z ní je dokazatelná každá formule. Není-li T sporná (tj. existuje formule, která není z T dokazatelná), nazývá se ***bezesporná*** (konzistentní).

Ukažme některé dokazatelné formule a příklady důkazů.

Věta 2.27 *Pro každou formuli φ platí $\vdash \varphi \Rightarrow \varphi$ (tj. formule $\varphi \Rightarrow \varphi$ je dokazatelná v našem axiomatickém systému).*

Důkaz. Máme ukázat, že existuje důkaz (z prázdné množiny T), jehož posledním prvkem je $\varphi \Rightarrow \varphi$. Důkazem je např. posloupnost formulí

$\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ [1. formule důkazu, tato formule je axiomem dle (1)]

$(\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ [2. formule důkazu, tato formule je axiomem dle (2)]

$(\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ [3. formule důkazu, tato formule vyplývá z předchozích dvou pomocí MP]

$\varphi \Rightarrow (\varphi \Rightarrow \varphi)$ [4. formule důkazu, tato formule je axiomem dle (1)]

$\varphi \Rightarrow \varphi$ [5. formule důkazu, dokazovaná formule, vyplývá ze dvou předchozích formulí pomocí MP]. □

Následující věta umožňuje mimo jiné zkracovat důkazy.

Věta 2.28 (o dedukci) *Pro každou množinu T formulí a formule φ, ψ platí*

$$T, \varphi \vdash \psi \text{ právě když } T \vdash \varphi \Rightarrow \psi.$$

Důkaz. "$\Leftarrow$": Předpokládáme-li $T \vdash \varphi \Rightarrow \psi$, je tím spíše $T, \varphi \vdash \varphi \Rightarrow \psi$. Použitím MP okamžitě dostáváme $T, \varphi \vdash \psi$.

"$\Rightarrow$": Nechť $T, \varphi \vdash \psi$, tj. existuje důkaz $\psi_1, \dots, \psi_n$ formule ψ z T, φ ($\psi_n = \psi$). Indukcí dokážeme, že $T \vdash \varphi \Rightarrow \psi_i$ platí pro $i = 1, \dots, n$, z čehož plyne požadovaný vztah (je speciálním případem pro $i = n$). Vezměme tedy $i \in \{1, \dots, n\}$ a předpokládejme, že pro každé $j < i$ je $T \vdash \varphi \Rightarrow \psi_j$ (indukční předpoklad). Dokážeme, že $T \vdash \varphi \Rightarrow \psi_i$. Podle definice důkazu mohou nastat pouze následující případy:

- ψ_i je axiom nebo formule z T. Pak je posloupnost formulí

  $\psi_i \Rightarrow (\varphi \Rightarrow \psi_i)$ [axiom typu (1)]

  ψ_i [podle předpokladu axiom nebo formule z T]

  $\varphi \Rightarrow \psi_i$ [použití MP]

  důkazem formule $\varphi \Rightarrow \psi_i$ z T.

- ψ_i je formulí φ. Pak $T \vdash \varphi \Rightarrow \psi_i$ plyne z Věty 2.27.

- ψ_i plyne z předchozích formulí $\psi_j, \psi_k = \psi_j \Rightarrow \psi_i$ ($j, k < i$) pomocí MP. Dle indukčního předpokladu existuje důkaz $\alpha, \dots, \psi_j$ z T a důkaz $\beta, \dots, \psi_j \Rightarrow \psi_i$ z T. Přidáme-li k posloupnosti $\alpha, \dots, \psi_j, \beta, \dots, \psi_j \Rightarrow \psi_i$ formule

  $(\varphi \Rightarrow (\psi_j \Rightarrow \psi_i)) \Rightarrow ((\varphi \Rightarrow \psi_j) \Rightarrow (\varphi \Rightarrow \psi_i))$ [axiom typu 3]

  $(\varphi \Rightarrow \psi_j) \Rightarrow (\varphi \Rightarrow \psi_i)$ [použití MP]

  $\varphi \Rightarrow \psi_i$ [použití MP],

  dostaneme důkaz formule $\varphi \Rightarrow \psi_i$ z T.

Důkaz je hotov. □

Poznámka 2.29 V situacích, kdy máme ukázat, že platí $T \vdash \varphi$ (tj., že z T je dokazatelné φ), budeme postupovat dvěma způsoby (jde nám teď zejména o to, jak přehledně zapisovat argumenty prokazující $T \vdash \varphi$).

První způsob: Budeme postupně vypisovat formule (spolu se zdůvodněním a komentářem), které tvoří, tak jak jsou uvedeny za sebou, důkaz formule φ z T. Tímto způsobem postupujeme v důkazu Věty 2.27.

Druhý způsob: Tvrzení $T \vdash \varphi$ dokážeme tak, že uvedeme posloupnost postupně zdůvodňovaných tvrzení tvaru $T_i \vdash \varphi_i$, přičemž posledním bude dokazované tvrzení $\vdash \varphi$ (tj. $T_i = \emptyset$ a $\varphi_i = \varphi$). Tímto způsobem postupujeme v důkazu Věty 2.31.

Příklad 2.30 Ukažme použití věty o dedukci. Ukažme, že jestliže $T \vdash \varphi \Rightarrow \psi$ a $T \vdash \psi \Rightarrow \chi$, pak $T \vdash \varphi \Rightarrow \chi$ (tomuto tvrzení říkáme princip tranzitivity implikace). Skutečně, máme $T, \varphi \vdash \psi$ (dle věty o dedukci aplikované na $T \vdash \varphi \Rightarrow \psi$), dále $T, \varphi \vdash \chi$ (použitím MP), a konečně $T \vdash \varphi \Rightarrow \chi$ (věta o dedukci použitá na $T, \varphi \vdash \chi$).

Následují užitečné dokazatelné formule a vztahy.

Věta 2.31 *Pro formule φ, ψ platí*

- $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$,
- $\vdash \neg\neg\varphi \Rightarrow \varphi$,
- $\vdash \varphi \Rightarrow \neg\neg\varphi$,
- $\vdash (\varphi \Rightarrow \psi) \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$,
- $\vdash \varphi \Rightarrow (\neg\psi \Rightarrow \neg(\varphi \Rightarrow \psi))$

Důkaz. Uvedená tvrzení jsou tvaru $\vdash \varphi$. Každé z nich dokážeme tak, že uvedeme posloupnost postupně zdůvodňovaných tvrzení tvaru $T_i \vdash \varphi_i$, přičemž posledním bude dokazované tvrzení $\vdash \varphi$ (tj. $T_i = \emptyset$ a $\varphi_i = \varphi$).

"$\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$":
$\vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ [axiom (1): formule $\neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ je instancí axiomu (1); $\vdash \psi$ platí pro každý axiom ψ]
$(\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ [axiom (3)]
$\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ [princip tranzitivity implikace použitý na výše uvedená tvrzení].

"$\vdash \neg\neg\varphi \Rightarrow \varphi$":
$\vdash \neg\neg\varphi \Rightarrow (\neg\varphi \Rightarrow \neg\neg\neg\varphi)$ [právě dokázané tvrzení: použití právě dokázaného tvrzení pro $\varphi := \neg\varphi$, $\psi := \neg\neg\neg\varphi$]
$\neg\neg\varphi \vdash (\neg\varphi \Rightarrow \neg\neg\neg\varphi)$ [věta o dedukci: použití věty o dedukci pro $T := \emptyset$, $\varphi := \neg\neg\varphi$, $\psi := (\neg\varphi \Rightarrow \neg\neg\neg\varphi)$]
$(\neg\varphi \Rightarrow \neg\neg\neg\varphi) \Rightarrow (\neg\neg\varphi \Rightarrow \varphi)$ [axiom (3)]
$\neg\neg\varphi \vdash \neg\neg\varphi \Rightarrow \varphi$ [MP (resp. Pozorování 2.25) použité na předchozí dvě tvrzení]
$\neg\neg\varphi \vdash \varphi$ [věta o dedukci a fakt, že $\neg\neg\varphi \Rightarrow \varphi$ je ekvivalentní $\neg\neg\varphi, \neg\neg\varphi \vdash \varphi$]
$\vdash \neg\neg\varphi \Rightarrow \varphi$ [věta o dedukci]

"$\vdash \varphi \Rightarrow \neg\neg\varphi$":
$\vdash \neg\neg\neg\varphi \Rightarrow \neg\varphi$ [předchozí tvrzení]
$\vdash (\neg\neg\neg\varphi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \neg\neg\varphi)$ [axiom (3)]
$\vdash \varphi \Rightarrow \neg\neg\varphi$ [MP]

"$\vdash (\varphi \Rightarrow \psi) \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$":
$\vdash \neg\neg\varphi \Rightarrow \varphi$, $\vdash \psi \Rightarrow \neg\neg\psi$ [předchozí tvrzení]

$(\varphi \Rightarrow \psi) \vdash (\neg\neg\, \varphi \Rightarrow \neg\neg\, \psi)$ [princip tranzitivity implikace (dvakrát)]
$(\neg\neg\, \varphi \Rightarrow \neg\neg\, \psi) \Rightarrow (\neg\, \psi \Rightarrow \neg\, \varphi)$ [axiom (3)]
$(\varphi \Rightarrow \psi) \vdash (\neg\, \psi \Rightarrow \neg\, \varphi)$ [MP]
$\vdash (\varphi \Rightarrow \psi) \Rightarrow (\neg\, \psi \Rightarrow \neg\, \varphi)$ [věta o dedukci]

"$\vdash \varphi \Rightarrow (\neg\, \psi \Rightarrow \neg\, (\varphi \Rightarrow \psi))$":
$\varphi, \varphi \Rightarrow \psi \vdash \psi$ [MP]
$\varphi \vdash (\varphi \Rightarrow \psi) \Rightarrow \psi$ [věta o dedukci]
$\varphi \vdash ((\varphi \Rightarrow \psi) \Rightarrow \psi \Rightarrow (\neg\, \psi \Rightarrow (\varphi \Rightarrow \psi)))$ [předchozí tvrzení]
$\varphi \vdash \neg\, \psi \Rightarrow (\varphi \Rightarrow \psi))$ [MP]
$\vdash \varphi \Rightarrow \neg\, \psi \Rightarrow (\varphi \Rightarrow \psi))$ [věta o dedukci]

□

Jsou-li φ formule, $p_1, \ldots, p_n$ po dvou různé výrokové symboly a $\varphi_1, \ldots, \varphi_n$ formule, označíme symbolem $\varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ formuli, která vznikne z formule φ nahrazením všech výskytů symbolů $p_1, \ldots, p_n$ po řadě formulemi $\varphi_1, \ldots, \varphi_n$. Je zřejmé, že $\varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ je skutečně formule.

Věta 2.32 (o nahrazení) *Pro libovolné formule $\varphi, \varphi_1, \ldots, \varphi_n$ a libovolné po dvou různé výrokové symboly $p_1, \ldots, p_n$ platí, že z $\vdash \varphi(p_1, \ldots, p_n)$ plyne $\vdash \varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$.*

Důkaz. Důkaz je jednoduchý. Předpokládejme, že $\vdash \varphi$. Je-li $\theta_1, \ldots, \theta_l$ důkaz formule φ (tj. $\theta_l = \varphi$), stačí dokázat, že $\theta_1(p_1/\varphi_1, \ldots, p_n/\varphi_n), \ldots, \theta_l(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ je také důkazem (to ověříme), a sice důkazem formule $\varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ (to je zřejmé, protože $\theta_l = \varphi$). Zbývá tedy ukázat, že $\theta_1(p_1/\varphi_1, \ldots, p_n/\varphi_n), \ldots, \theta_l(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ je důkaz. Musíme ověřit, že uvedená posloupnost formulí splňuje podmínky definice důkazu. Uvažujme libovolný člen $\theta_i(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ této posloupnosti. Protože původní posloupnost $\theta_1, \ldots, \theta_l$ je důkazem, nastávají pro θ_i následující dvě možnosti.

První: θ_i je axiom. Pak je ale axiomem také $\theta_i(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ (proč?).

Druhá: θ_i vznikne použitím MP na nějaké $\theta_j, \theta_k,\ j, k < i$. Pak ale vznikne $\theta_i(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ použitím MP na $\theta_j(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ a $\theta_k(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ (proč?).

Vidíme tedy, že každý člen $\theta_i(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ je buď axiomem nebo vznikne použitím MP na předcházející členy, a tedy uvedená posloupnost je důkazem. □

Poznámka 2.33 Věta 2.32 tedy říká, že je-li φ dokazatelná formule, jejíž všechny výrokové symboly jsou mezi $p_1, \ldots, p_n$, pak nahradíme-li všechny výskyty symbolů p_i formulemi φ_i, takto vzniklá formule $\varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ je také dokazatelná. Všimněme si, že platí i obecnější tvrzení: Jsou-li $q_1, \ldots, q_k \in \{p_1, \ldots, p_n\}$ výrokové symboly, které se vyskytují v φ (ale nemusí to být všechny, φ může obsahovat i jiné než q_j), pak nahradíme-li všechny výskyty proměnných $q_1, \ldots, q_k$ po řadě formulemi $\varphi_1, \ldots, \varphi_n$ a označíme-li takto vzniklou formuli ψ, pak z dokazatelnosti φ plyne dokazatelnost ψ. Proč tomu tak je? [nápověda: Ve formuli $\varphi(p_1/\varphi_1, \ldots, p_n/\varphi_n)$ mohou být některé φ_i rovny přímo p_i, a tedy jako by probíhala substituce jen za některé p_i]

Věta 2.34 (o ekvivalenci) *Vznikne-li formule ψ a formule φ nahrazením jejích podformulí $\varphi_1, \ldots, \varphi_n$ po řadě formulemi $\psi_1, \ldots, \psi_n$, pak*

$$\varphi_1 \Rightarrow \psi_1, \psi_1 \Rightarrow \varphi_1, \ldots, \chi_n \Rightarrow \varphi_n, \psi_n \Rightarrow \varphi_n \vdash \varphi \Rightarrow \psi.$$

Z $\vdash \varphi$ tedy plyne $\varphi_1 \Rightarrow \psi_1, \psi_1 \Rightarrow \varphi_1, \ldots, \chi_n \Rightarrow \varphi_n, \psi_n \Rightarrow \varphi_n \vdash \psi$.

Důkaz. Viz přednášky. □

Věta 2.35 (o důkazu sporem) *Pro teorii T a formule φ a ψ je $T \vdash \varphi$, právě když $T, \neg\, \varphi \vdash \neg\, (\psi \Rightarrow \psi)$.*

Důkaz. "$\Rightarrow$": Předpokládejme $T \vdash \varphi$. Máme
$\vdash \neg\, \varphi \Rightarrow (\varphi \Rightarrow \neg\, (\psi \Rightarrow \psi))$ [Věta 2.31]
$T, \neg\, \varphi \vdash (\varphi \Rightarrow \neg\, (\psi \Rightarrow \psi))$ [věta o dedukci]
$T, \neg\, \varphi \vdash \neg\, (\psi \Rightarrow \psi)$ [MP na předchozí vztah a na předpoklad $T \vdash \varphi$].

"$\Leftarrow$": Předpokládejme $T, \neg\, \varphi \vdash \neg\, (\psi \Rightarrow \psi)$. Máme
$T \vdash \neg\, \varphi \Rightarrow \neg\, (\psi \Rightarrow \psi)$ [věta o dedukci]
$\vdash [\neg\, \varphi \Rightarrow \neg\, (\psi \Rightarrow \psi)] \Rightarrow [(\psi \Rightarrow \psi) \Rightarrow \varphi]$ [axiom (3)]
$T \vdash (\psi \Rightarrow \psi) \Rightarrow \varphi$ [MP]
$\vdash \psi \Rightarrow \psi$ [Věta 2.27]
$T \vdash \varphi$ [MP]. □

Věta 2.36 (o důkazu rozborem případů) *Pro teorii T a formule φ, ψ, χ platí $T, \varphi \vee \psi \vdash \chi$ právě když $T, \varphi \vdash \chi$ a $T, \psi \vdash \chi$.*

Důkaz. Předpokládejme $T, \varphi \vee \psi \vdash \chi$, tj. $T, \neg\, \varphi \Rightarrow \psi \vdash \chi$. Máme $\varphi \Rightarrow (\neg\, \varphi \Rightarrow \psi)$ (Věta 2.31), použitím MP tedy $T, \varphi \vdash \neg\, \varphi \Rightarrow \psi$, což dále s použitím předpokladu $T, \neg\, \varphi \Rightarrow \psi \vdash \chi$ dává $T, \varphi \vdash \chi$. Fakt $T, \psi \vdash \chi$ dostaneme obdobně použitím $\psi \Rightarrow (\neg\, \varphi \Rightarrow \psi)$ (axiom (1)).

Předpokládejme $T, \varphi \vdash \chi$ a $T, \psi \vdash \chi$. Máme
$T \vdash \varphi \Rightarrow \chi$ [věta o dedukci]
$T \vdash \psi \Rightarrow \chi$ [věta o dedukci]
$T \vdash (\varphi \Rightarrow \chi) \Rightarrow (\neg\, \chi \Rightarrow \neg\, \varphi)$ [Věta 2.31]
$T \vdash \neg\, \chi \Rightarrow \neg\, \varphi$ [MP]
$T, \neg\, \varphi \Rightarrow \psi \vdash \neg\, \chi \Rightarrow \chi$ [princip tranzitivity implikace: máme $T \vdash \neg\, \chi \Rightarrow \neg\, \varphi$, v předpokladu $\neg\, \varphi \Rightarrow \psi$, $T \vdash \psi \Rightarrow \chi$]
$\vdash \neg\, \chi \Rightarrow (\neg\, \chi \Rightarrow \neg\, (\neg\, \chi \Rightarrow \chi))$ [Věta 2.31 použitá na $\neg\, \chi$ a χ]
$\vdash \neg\, \chi \Rightarrow \neg\, (\neg\, \chi \Rightarrow \chi)$ [3× věta o dedukci (dvakrát přesun vlevo, jednou vpravo)]
$(\neg\, \chi \Rightarrow \chi) \Rightarrow \chi$ [axiom (3) a MP na předchozí dokazatelnost]
$T, \neg\, \varphi \Rightarrow \psi \vdash \chi$ [MP na předchozí],
čímž je tvrzení dokázané, neboť $\varphi \vee \psi$ je zkratkou za $\neg\, \varphi \Rightarrow \psi$. □

Věta 2.37 (o neutrální formuli) *Pro teorii T a formule φ a ψ platí $T \vdash \psi$, právě když $T, \varphi \vdash \psi$ a $T, \neg\varphi \vdash \psi$.*

Důkaz. Podle Věty o důkazu rozborem případů je $T, \varphi \vee \neg\varphi \vdash \psi$, právě když $T, \varphi \vdash \psi$ a $T, \neg\varphi \vdash \psi$. Dále však platí, že $T, \varphi \vee \neg\varphi \vdash \psi$, právě když $T \vdash \psi$ (neboť $\varphi \vee \neg\varphi$ je zkratkou za $\neg\varphi \Rightarrow \neg\varphi$, což je dokazatelná formule dle Věty 2.27; pro dokazatelnou α je vždy $T, \alpha \vdash \beta$, právě když $T \vdash \beta$), a tím je důkaz hotov. □

Viděli jsme formule, které jsou v našem axiomatickém systému dokazatelné. V následujícím se lehce přesvědčíme, že každá dokazatelná formule je tautologií. Vzniká otázka, zda také naopak je každá tautologie dokazatelná. Uvidíme, že ano (a uvidíme i více). Jinými slovy, naše axiomy a odvozovací pravidlo (obojí je velmi jednoduché) jsou zvoleny tak vhodně, že umožňují dokázat všechny tautologie, ale žádné další formule (tj. formule, které jsou někdy nepravdivé).

Věta 2.38 (o korektnosti) *Pro libovolnou množinu T formulí a formuli φ platí, že je-li $T \vdash \varphi$, pak $T \models \varphi$. Speciálně tedy, každá dokazatelná formule je tautologií.*

Důkaz. Nejprve pro $T = \emptyset$. Každý axiom je tautologie (přesvědčte se např. tabulkovou metodou). Dále platí, že jsou-li φ a $\varphi \Rightarrow \psi$ tautologie, je i ψ tautologie (je ihned vidět, ověřte). Indukcí tedy dostáváme, že každý člen důkazu je tautologie. Tedy každá dokazatelná formule je tautologie.

Je-li $T \neq \emptyset$, pak z $T \vdash \varphi$ plyne, že pro nějaké $\psi_1, \ldots, \psi_n \in T$ je $\psi_1, \ldots, \psi_n \vdash \varphi$. n-násobným použitím věty o dedukci odtud dostaneme $\vdash \psi_1 \Rightarrow (\psi_2 \Rightarrow (\cdots(\psi_n \Rightarrow \varphi)\cdots))$, z čehož dle výše dokázaného plyne $\models \psi_1 \Rightarrow (\psi_2 \Rightarrow (\cdots(\psi_n \Rightarrow \varphi)\cdots))$. Nyní n-násobně použijeme "sémantické verze" věty o dedukci (viz Příklad 2.21) a dostaneme $\psi_1, \ldots, \psi_n \models \varphi$, z čehož plyne (uvědomte si proč) $T \models \varphi$. □

Před důkazem věty o úplnosti zavedeme následující značení. Pro formuli φ a ohodnocení e je

$$\varphi^e = \begin{cases} \varphi & \text{pokud } \|\varphi\|_e = 1 \\ \neg\varphi & \text{pokud } \|\varphi\|_e = 0. \end{cases}$$

Lemma 2.39 *Pro libovolnou formuli φ, která neobsahuje jiné výrokové symboly než $p_1, \ldots, p_n$, platí*

$$p_1^e, \ldots, p_n^e \vdash \varphi^e.$$

Důkaz. Tvrzení dokážeme indukcí přes složitost formule φ.

Nechť φ je výrokový symbol p. Pak je tvrzení zřejmé.

Nechť tvrzení platí pro φ. Ukažme, že pak platí i pro $\neg\varphi$, tj. že $p_1^e, \ldots, p_n^e \vdash (\neg\varphi)^e$. Rozlišme dva případy, $\|\varphi\|_e = 0$ a $\|\varphi\|_e = 1$. Pro $\|\varphi\|_e = 0$ je $\varphi^e = \neg\varphi$ a $(\neg\varphi)^e = \neg\varphi$. Požadované tvrzení $p_1^e, \ldots, p_n^e \vdash (\neg\varphi)^e$ tedy přímo plyne z předpokladu. Pro $\|\varphi\|_e = 1$ je $\varphi^e = \varphi$ a $(\neg\,\varphi)^e = \neg\neg\,\varphi$. Podle předpokladu je tedy $p_1^e, \ldots, p_n^e \vdash \varphi$ a máme dokázat $p_1^e, \ldots, p_n^e \vdash \neg\neg\,\varphi$. To však plyne z $p_1^e, \ldots, p_n^e \vdash \varphi$ a z $\vdash \varphi \Rightarrow \neg\neg\,\varphi$ (viz Věta 2.31) pomocí MP.

Nechť tvrzení platí pro φ a ψ. Ukažme, že pak platí i pro $\varphi \Rightarrow \psi$, tj. že $p_1^e, \ldots, p_n^e \vdash (\varphi \Rightarrow \psi)^e$. Mohou nastat následující případy:

- $\|\varphi\|_e = 0$: Pak je $\|\varphi \Rightarrow \psi\|_e = 1$, tedy $(\varphi \Rightarrow \psi)^e = \varphi \Rightarrow \psi$. Podle předpokladu máme $p_1^e, \ldots, p_n^e \vdash \neg\,\varphi$. Podle Věty 2.31 je $\vdash \neg\,\varphi \Rightarrow (\varphi \Rightarrow \psi)$, odkud pomocí MP dostaneme požadované $p_1^e, \ldots, p_n^e \vdash \varphi \Rightarrow \psi$.

- $\|\psi\|_e = 1$: Pak je $\|\varphi \Rightarrow \psi\|_e = 1$, tedy opět $(\varphi \Rightarrow \psi)^e = \varphi \Rightarrow \psi$. Podle předpokladu máme $p_1^e, \ldots, p_n^e \vdash \psi$. Z (1) a MP dostaneme požadované $p_1^e, \ldots, p_n^e \vdash \varphi \Rightarrow \psi$.

- $\|\varphi\|_e = 1$ a $\|\psi\|_e = 0$: Pak $\|\varphi \Rightarrow \psi\|_e = 0$, tedy $(\varphi \Rightarrow \psi)^e = \neg\,(\varphi \Rightarrow \psi)$. Podle předpokladu je $p_1^e, \ldots, p_n^e \vdash \varphi$ a $p_1^e, \ldots, p_n^e \vdash \neg\,\psi$. Použitím Věty 2.31 (poslední uvedená dokazatelnost) a dvojnásobnýmpoužitím MP dostaneme požadované $p_1^e, \ldots, p_n^e \vdash \neg\,(\varphi \Rightarrow \psi)$.

□

Věta 2.40 (o úplnosti, slabá verze) *Pro libovolnou konečnou množinu T formulí a formuli φ platí, že z $T \models \varphi$ plyne $T \vdash \varphi$. Speciálně, každá pravdivá formule je dokazatelná.*

Důkaz. Tvrzení dokážeme nejdříve pro případ $T = \emptyset$. Nechť tedy $\models \varphi$. Pro každé ohodnocení e tedy platí $\varphi^e = \varphi$ (protože podle předpokladu je $\|\varphi\|_e = 1$). Jsou-li $p_1, \ldots, p_n$ všechny výrokové symboly z φ, je tedy dle Lemma 2.39

$$p_1^e, p_2^e, \ldots, p_n^e \vdash \varphi.$$

Uvažujme nyní ohodnocení e', které se o e liší právě v hodnotě, kterou přiřazuje symbolu p_1. Předpokládejme, že $e(p_1) = 1$ a $e'(p_1) = 0$ (případ $e(p_1) = 0$ a $e'(p_1) = 1$ se ošetří symetricky). Dle Lemma 2.39 je opět

$$p_1^{e'}, p_2^{e'}, \ldots, p_n^{e'} \vdash \varphi.$$

Protože je však podle předpokladu $p_2^e = p_2^{e'}, \ldots, p_n^e = p_n^{e'}$, $p_1^e = p_1$ a $p_1^{e'} = \neg\,p_1$, dostáváme

$$p_1, p_2^e, \ldots, p_n^e \vdash \varphi.$$

a

$$\neg\,p_1, p_2^e, \ldots, p_n^e \vdash \varphi,$$

odkud podle Věty 2.37 máme

$$p_2^e, \ldots, p_n^e \vdash \varphi.$$

Opakovaným použitím právě provedené úvahy postupně dostaneme

$$p_3^e, \ldots, p_n^e \vdash \varphi$$

až po

$$p_n^e \vdash \varphi$$

a nakonec

$$\vdash \varphi.$$

Nechť nyní $T = \{\varphi_1, \ldots, \varphi_n\}$. Podle sémantické verzi věty o dedukci dostaneme z $T \models \varphi$, že $\models \varphi_1 \Rightarrow (\cdots(\varphi_n \Rightarrow \varphi))$. Odtud podle právě dokázaného plyne $\vdash \varphi_1 \Rightarrow (\cdots(\varphi_n \Rightarrow \varphi))$, odkud dostáváme pomocí věty o dedukci $\varphi_1, \ldots, \varphi_n \vdash \varphi$, tj. požadované $T \vdash \varphi$. □

Pro důkaz tzv. silné verze věty o úplnosti (ta se neomezuje na konečné T) potřebujeme následující větu.

Věta 2.41 (o kompaktnosti) *(1) Množina T formulí je splnitelná, právě když je splnitelná každá konečná podmnožina množiny T. (2) Pro každou formuli φ je $T \models \varphi$, právě když existuje konečná $S \subseteq T$ tak, že $S \models \varphi$.*

Důkaz. (nebude u zkoušky požadován) □

S použitím Věty 2.41 snadno dokážeme následující větu.

Věta 2.42 (o úplnosti, silná verze) *Pro libovolnou množinu T formulí a formuli φ platí, že z $T \models \varphi$ plyne $T \vdash \varphi$.*

Důkaz. Je-li $T \models \varphi$, pak dle Věty 2.41 (2) existuje konečná $S \subseteq T$ tak, že $S \models \varphi$. Podle Věty 2.40 je $S \vdash \varphi$, a z toho samozřejmě plyne $T \vdash \varphi$. □

Uvědomme si, že Věta o úplnosti je velmi netriviální tvrzení: Z toho, že nějaká formule má při všech (intuitivně zcela přirozeně definovaných) možných ohodnoceních pravdivostní hodnotu 1 plyne, že je dokazatelná pomocí tří (jednoduchých a intuitivně přijatelných) axiomů a jednoho (jednoduchého a intuitivně přijatelného) odvozovacího pravidla. Pojem pravdivého tvrzení, tak jak je formalizován v rámci výrokové logiky, je tedy plně syntakticky charakterizovatelný (a navíc velmi jednoduchým zůsobem).

Následující věta ukazuje další vztah dvojice pojmů, jednoho sémantického (splnitelnost) a druhého syntaktického (bezespornost), které spolu na první pohled nesouvisí.

Věta 2.43 *Množina T formulí je splnitelná, právě když je bezesporná.*

Důkaz. Nechť je T splnitelná. Pak existuje ohodnocení e, ve kterém jsou pravdivé všechny formule z T. Kdyby byla T sporná, pak by pro libovolnou formuli φ bylo $T \vdash \varphi$ a $T \vdash \neg \varphi$, a tedy dle korektnosti i $T \models \varphi$ a $T \models \neg \varphi$. To znamená, že při každém ohodnocení, při kterém jsou pravdivé všechny formule z T (jedním z nich je e), je pravdivá jak formule φ, tak formule $\neg \varphi$. To je ale pochopitelně nemožné.

Nechť T je bezesporná. Pak existuje formule φ, pro kterou neplatí $T \vdash \varphi$, tj. (podle úplnosti) neplatí $T \models \varphi$. To ale znamená, že existuje ohodnocení, ve kterém není pravdivá φ, ale jsou pravdivé všechny formule z T. Tedy T je splnitelná. $\square$

3 Predikátová logika

3.1 Syntax predikátové logiky

Přirozený jazyk je základním prostředkem, pomocí kterého formulujeme a zaznamenáváme své usuzování. Již základní rozbor vět přirozeného jazyka odhalí některé jeho významné jednotky/součásti. Pro nás to budou: proměnné, relační symboly (symboly pro označování relací), funkční symboly (symboly pro označování funkcí) včetně symbolů pro označování konstant , symboly pro logické spojky (jako "a", "nebo", "právě když" atd.), symboly pro kvantifikátory (např. "pro každý", "existuje") a pomocné symboly.

Ve větách "Každý slon je savec", "Pro každé dva body existuje bod, který mezi nimi neleží", "Petr je mladší než Pavel nebo věk Jiřího je větší než součet věků Petra a Pavla", "Součet druhých mocnin nenulových čísel je větší než nula" se mluví o jednomístným vztazích "být slonem", "být savcem", trojmístném vztahu "neležet mezi dvěma body", dvojmístném vztahu "být větší", o funkci přiřazující člověku jeho věk, o funkci sčítání, o funkci mocnění, o (konstantních) objektech Petr, Pavel, Jiří, nula, implicitně se zde objevují proměnné (např. první tvrzení, formulováno přesněji, říká "pro každé x platí, že je-li x slonem, je x savcem"), logické spojky ("nebo") a kvantifikátory ("pro každé", "existuje").

Definice 3.1 ***Jazyk $\mathcal{J}$ predikátové logiky*** obsahuje (a je tím určen)

- ***(předmětové) proměnné*** $x, y, z, \ldots, x_1, x_2, \ldots$; předpokládáme, že jich je nekonečně mnoho (spočetně mnoho, protože chceme mít jazyk spočetný);
- ***relační symboly*** $p, q, r, \ldots, r_1, r_2, \ldots$, ke každému relačnímu symbolu r je dáno nezáporné celé číslo $\sigma(r)$ nazývané *arita* symbolu r; musí existovat aspoň jeden relační symbol;
- ***funkční symboly*** $f, g, h, \ldots, f_1, f_2, \ldots$, ke každému funkčnímu symbolu f je dáno nezáporné celé číslo $\sigma(f)$ nazývané *arita* symbolu f;
- symboly pro ***logické spojky*** $\neg$ (negace) a $\Rightarrow$ (implikace) a symbol $\forall$ pro univerzální kvantifikátor;

- ***pomocné symboly*** — různé typy závorek.

Poznámka 3.2 (1) Místo předmětové proměnné často říkáme jen proměnné. Množina všech relačních (někdy se říká predikátových) symbolů jazyka $\mathcal{J}$ se značí R (popř. $R_{\mathcal{J}}$); množina všech funkčních (někdy se říká operačních) symbolů jazyka $\mathcal{J}$ se značí F (popř. $F_{\mathcal{J}}$). Je-li $r \in R$ a $\sigma(r) = n$, pak říkáme, že r je n-árni; podobně pro $f \in F$. Je-li $f \in F$ 0-ární, nazývá se f symbol konstanty (neboť funkce, která má 0 argumentů, musí přiřazovat vždy stejnou hodnotu, tj. je konstantní; zařadit cvičení na vysvětlení).

(2) Je zřejmé, že jazyk je jednoznačně určen svými relačními symboly, funkčními symboly a jejich aritami (vše ostatní mají všechny jazyky stejné). Trojici $\langle R, J, \sigma \rangle$ proto nazýváme *typ jazyka*.

(3) Je-li mezi relačními symboly symbol $\approx$, nazýváme ho symbol pro rovnost a celý jazyk pak *jazyk s rovností*. Symbol pro rovnost a rovnost má specifické postavení, jak uvidíme dále.

Základní syntaktické jednotky vybudované ze symbolů jazyka jsou termy a formule. Termy jsou výrazy reprezentující funkci aplikovanou na své operandy; formule reprezentují tvrzení o prvcích univerza.

Definice 3.3 ***Term*** jazyka typu $\langle R, F, \sigma \rangle$ je induktivně definován takto:

(i) každá proměnná x je term;

(ii) je-li $f \in F$ n-ární a jsou-li $t_1, \dots, t_n$ termy, pak $f(t_1, \dots, t_n)$ je term.

Poznámka 3.4 (1) Termy jsou tedy jisté konečné posloupnosti prvků daného jazyka.

(2) Je-li $f \in F$ binární, používáme také tzv. infixovou notaci a píšeme xfy nebo (xfy) místo $f(x, y)$ (např. $2 + 3$ místo $+(2, 3)$); ve složených termech používáme závorky (např. $(2 + 3) \cdot 5$).

Definice 3.5 ***Formule*** jazyka typu $\langle R, F, \sigma \rangle$ je induktivně definována takto:

(i) je-li $r \in R$ n-ární a $t_1, \dots, t_n$ jsou termy, pak $r(t_1, \dots, t_n)$ je formule;

(ii) jsou-li φ a ψ formule, pak $\neg\, \varphi$, $(\varphi \Rightarrow \psi)$ jsou také formule;

(iii) je-li φ formule a x proměnná, pak $(\forall x)\varphi$ je formule.

Poznámka 3.6 (1) Formule vytvořené dle (i) se nazývají *atomické*. Je-li $r \in R$ binární, píšeme také $t_1 r t_2$ nebo $(t_1 r t_2)$ místo $r(t_1, t_2)$ (tedy např. $x \leq y$ místo $\leq (x, y)$). Obzvláště píšeme $t_1 \approx t_2$ místo $\approx (t_1, t_2)$.

(2) Budeme používat obvyklé konvence, zjednodušující čitelnost formulí: vynechávání závorek, píšeme $\dots (\forall x, y) \dots$ místo $\dots (\forall x)(\forall y) \dots$ atd.).

(3) Podobně jako ve výrokové logice, symboly pro ostatní logické spojky nepatří do jazyka predikátové logiky. Podobně symbol $\exists$ (existenční kvantifikátor)

nepatří do jazyka predikátové logiky. Abychom tyto symboly měli k dispozici, chápeme

$$\text{posloupnost } \varphi \wedge \psi \text{ jako zkratku za } \neg(\varphi \Rightarrow \neg\psi), \tag{4}$$

$$\text{posloupnost } \varphi \vee \psi \text{ jako zkratku za } \neg\varphi \Rightarrow \psi, \tag{5}$$

$$\text{posloupnost } \varphi \Leftrightarrow \psi \text{ jako zkratku za } (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi), \tag{6}$$

$$\text{posloupnost } (\exists x)\varphi \text{ jako zkratku za } \neg(\forall x)\neg\varphi. \tag{7}$$

Posloupnosti obsahující symboly $\wedge, \vee, \Leftrightarrow$ a $\exists$ tedy nejsou formulemi, mohou to být zkratky za formule (příklad...). Pro jednoduchost však vědomi si toho, že a jakým způsobem se dopouštíme nepřesnosti, budeme těmto posloupnostem často také říkat formule.

Podobně jako ve výrokové logice můžeme i v predikátové logice provádět důkazy strukturální indukcí.

Věta 3.7 (důkaz strukturální indukcí pro termy) *Nechť $\mathcal{V}$ je vlastnost termů. Nechť platí, že*

- *každá proměnná má vlastnost $\mathcal{V}$;*
- *mají-li termy $t_1, \dots, t_n$ vlastnost $\mathcal{V}$ a je-li $f \in F$ n-ární, pak vlastnost $\mathcal{V}$ má i term $f(t_1, \dots, t_n)$.*

Pak vlastnost $\mathcal{V}$ má každý term.

Věta 3.8 (důkaz strukturální indukcí pro formule) *Nechť $\mathcal{V}$ je vlastnost formulí. Nechť platí, že*

- *každá atomická formule má vlastnost $\mathcal{V}$;*
- *mají-li formule φ a ψ vlastnost $\mathcal{V}$, pak vlastnost $\mathcal{V}$ mají i formule $\neg\varphi$ a $(\varphi \Rightarrow \psi)$;*
- *má-li formule φ $\mathcal{V}$, má vlastnost $\mathcal{V}$ formule $(\forall)\varphi$.*

Pak vlastnost $\mathcal{V}$ má každá formule.

Příklad 3.9 (1) Uvažujme jazyk $\mathcal{J}$ typu $\langle R, F, \sigma \rangle$, kde $R = \{p, d, b, s\}$, $F = \emptyset$, relační symboly p, d, b jsou unární, s binární. Je to jazyk bez funkčních symbolů, a tedy jedinými termy jsou proměnné. Atomickými formulemi jsou $p(x)$, $p(y)$, $d(y)$, $b(x)$, atd. Dalšími formulemi (ne atomickými) jsou např. výrazy $p(x) \vee p(x)$, $p(x) \vee\neg p(x)$, $(\forall x)(p(x) \wedge\neg d(x) \Rightarrow b(x))$, $(\exists x)b(x) \wedge\neg p(x)$, $(\forall x)(\forall y)(b(x) \Rightarrow (s(x,y) \Rightarrow b(y)))$.

Výrazy $((x \Rightarrow$, $p(x,x)$, $p(d(x))$, $(\forall x)(s(x,y) \Rightarrow\Rightarrow p(x))$, formulemi nejsou.

(2) Uvažujme jazyk $\mathcal{J}$ typu $\langle R, F, \sigma \rangle$, kde $R = \{p, \leq\}$, $F = \{c, \circ\}$, c je nulární (tj. symbol konstanty), p je unární, $\leq$ a $\circ$ jsou binární. Pak výrazy c, x, $c \circ c$, $c \circ (x \circ y)$, atd. jsou termy. Výrazy cc, $c \circ p(x)$, $p(x)$, $c \circ \circ x$, termy nejsou. Formulemi jsou např. $c \leq x$, $p(x) \Rightarrow (c \leq x)$, $(\forall x)(x \leq x \circ x)$, $(\forall x)(\forall y)(x \circ y \leq y \circ x)$.

Poznámka 3.10 Jazyk predikátové logiky obsahuje symboly různých typů (relační symboly, funkční symboly, symboly spojek). Z nich se dají vytvářet termy a formule, které jsou v určitém smyslu "rozumnými" řetězci. Rozumné proto, že dáme-li relačním a funkčním symbolům smysl, dají se "rozumně" číst (poznamenejme, že přesný smysl dostanou relační a funkční symboly, a také termy a formule, až vybudujeme sémantiku). Tak například, uvažujme jazyk z Příkladu 3.9 (1). Nechť p, d, b, m označují po řadě "mít dostatečný příjem", "mít velké dluhy", "být bonitní", "být manželé", tj. $p(x)$ znamená "objekt označený x má dostatečný příjem" atd. Formule $(\forall x)(p(x) \wedge\neg\, d(x) \Rightarrow b(x))$ pak "říká": "pro každé x platí, že má-li x dostatečný příjem a nemá-li velké dluhy, pak je x bonitní". Formule $(\forall x)(\forall y)(b(x) \Rightarrow (m(x,y) \Rightarrow b(y)))$ "říká": "pro každé x a y platí, že je-li x bonitní a jsou-li x a y manželé, je i y bonitní".

Můžeme také postupovat obráceně (to je dobré cvičení): K dané větě přirozeného jazyka navrhneme jazyk predikátové logiky a napíšeme formuli, která odpovídá danému tvrzení (viz cvičení).

Termy a formule jsou definovány induktivně. Každý term použitý při konstrukci termu t se nazývá ***podterm*** termu t; každá formule použitá v konstrukci formule φ se nazývá ***podformule*** formule φ. Přesněji: Množina $\mathrm{sub}(t)$ všech podtermů termu t je definována následovně: pro proměnnou x je $\mathrm{sub}(x) = \{x\}$; $\mathrm{sub}(f(t_1,\dots,t_n)) = \{f(t_1,\dots,t_n)\} \cup \mathrm{sub}(t_1) \cup \dots \cup \mathrm{sub}(t_n)$. Množina $\mathrm{sub}(\varphi)$ všech podformulí formule φ je definována následovně: je-li φ atomická, pak $\mathrm{sub}(\varphi) = \{\varphi\}$; $\mathrm{sub}(\neg\,\varphi) = \{\neg\,\varphi\} \cup \mathrm{sub}(\varphi)$ a $\mathrm{sub}(\varphi \Rightarrow \psi) = \{\varphi \Rightarrow \psi\} \cup \mathrm{sub}(\varphi) \cup \mathrm{sub}(\psi)$; $\mathrm{sub}((\forall x)\varphi) = \{(\forall x)\varphi\} \cup \mathrm{sub}(\varphi)$.

Říkáme, že proměnná ***se vyskytuje*** v termu nebo ve formuli, jestliže je některým symbolem termu nebo formule jakožto řetězce symbolů s tou výjimkou, že výskyty za $\forall$ se nepočítají (tj. proměnná x nemá výskyt ve formuli $(\forall x)r(y,z)$. Množinu všech proměnných, které se vyskytují v termu t označujeme $\mathrm{var}(t)$; množinu všech proměnných, které se vyskytují ve formuli φ označujeme $\mathrm{var}(\varphi)$.

Je-li t term, píšeme $t(x_1,\dots,x_n)$, abychom zdůraznili, že všechny proměnné, které se v t vyskytují, se nacházejí mezi $x_1,\dots,x_n$.

Proměnné z $\mathrm{var}(\varphi)$ mohou mít ve formuli φ výskyt dvojího druhu — volný a vázaný. Množina $\mathrm{free}(\varphi)$ proměnných, které mají ve φ ***volný výskyt*** je definována následovně: je-li φ atomická, pak $\mathrm{free}(\varphi) = \mathrm{var}(\varphi)$; dále $\mathrm{free}(\neg\,\varphi) = \mathrm{free}(\varphi)$, $\mathrm{free}(\varphi \Rightarrow \psi) = \mathrm{free}(\varphi) \cup \mathrm{free}(\psi)$; a $\mathrm{free}((\forall x)\varphi) = \mathrm{free}(\varphi) - \{x\}$. Množina $\mathrm{bound}(\varphi)$ proměnných, které mají ve φ ***vázaný výskyt*** je definována následovně: je-li φ atomická, pak $\mathrm{bound}(\varphi) = \emptyset$; dále $\mathrm{bound}(\neg\,\varphi) = \mathrm{bound}(\varphi)$, $\mathrm{bound}(\varphi \Rightarrow \psi) = \mathrm{bound}(\varphi) \cup \mathrm{bound}(\psi)$; a $\mathrm{bound}((\forall x)\varphi) = \mathrm{bound}(\varphi) \cup \{x\}$.

Píšeme $\varphi(x_1,\dots,x_n)$, abychom zdůraznili, že všechny proměnné, které mají ve φ volný výskyt, se nacházejí mezi $x_1,\dots,x_n$, tj. $\mathrm{free}(\varphi) \subseteq \{x_1,\dots,x_n\}$.

Poznámka 3.11 Chceme-li napsat formuli nebo term, musíme nejdříve popsat jazyk, jehož formuli nebo term chceme napsat. To je však často pracné a zbytečné. Proto zavedeme pojem indukovaného jazyka. ***Jazyk indukovaný*** množinami $\mathcal{F}$ a $\mathcal{T}$ řetězců je nejmenší jazyk $\mathcal{J}$ typu $\langle R, F, \sigma\rangle$, v němž řetězce z $\mathcal{F}$ formulemi a řetězce z $\mathcal{T}$ termy (tj. je-li $\mathcal{J}'$ jiný takový jazyk typu $\langle R', F', \sigma'\rangle$, pak $R \subseteq R'$, $F \subseteq F'$). Například jazykem určeným množinami $\mathcal{F} = ...$ a $\mathcal{T} = ...$ je ... . Místo jazyk indukovaný množinami $\mathcal{F}$ a $\mathcal{T}$ řetězců také říkáme jazyk určený formulemi z $\mathcal{F}$ a termy z $\mathcal{T}$.

Příklad 3.12 (Viz přednášky a cvičení) Uvažujme jazyk určený termy $t_1 = x_1 + ((c + x_1) + x_2)$, $t_2 = (c + c) + c$ and formulemi $\varphi_1 = (\forall x)(\forall y)((\mathrm{big}(x) \wedge (x \leq y)) \Rightarrow \mathrm{big}(y))$, $\varphi_2 = (c \leq x) \wedge (\forall x)(\exists y)(x + x \leq x + y)$. Pak $\mathrm{var}(\varphi_1) = \{x, y\}$, $\mathrm{var}(\varphi_2) = \{x, y\}$, $\mathrm{free}(\varphi_1) = \emptyset$, $\mathrm{bound}(\varphi_1) = \{x, y\}$, $\mathrm{free}(\varphi_2) = \{x\}$, $\mathrm{bound}(\varphi_2) = \{x, y\}$ (proměnná x má volný i vázaný výskyt v φ_2).

Užitečným pojmem je pojem substituce termu za proměnnou.

Definice 3.13 Nechť t a s jsou termy, x proměnná. ***Výsledek substituce*** termu s za x v t je term $t(x/s)$ definovaný následovně:

(i) je-li t proměnná, pak

$$t(x/s) = \begin{cases} s & \text{pro } t = x \\ t & \text{pro } t \neq x; \end{cases}$$

(ii) pro $t = f(t_1, \ldots, t_n)$, kde $f \in F$ je n-ární a $t_1, \ldots, t_n$ jsou termy, je $t(x/s) = f(t_1(x/s), \ldots, t_n(x/s))$.

Definice 3.14 Pro formuli φ, term s, a proměnnou x, je ***výsledkem substituce*** s za x ve φ formule $\varphi(x/s)$ definovaná následovně:

(i) pro $\varphi = r(t_1, \ldots, t_n)$ je $\varphi(x/s) = r(t_1(x/s), \ldots, t_n(x/s))$;

(ii) $(\neg\, \varphi)(x/s) = \neg\, (\varphi(x/s))$, $(\varphi \Rightarrow \psi)(x/s) = \varphi(x/s) \Rightarrow \psi(x/s)$;

(iii) $((\forall y)\varphi)(x/s) = (\forall y)\varphi$ pro $y = x$, $((\forall y)\varphi)(x/s) = (\forall y)(\varphi(x/s))$ pro $y \neq x$.

Příklad 3.15 K procvičení dokažte strukturální indukcí, že jsou-li t, s termy, φ formule a x proměnná, je $t(x/s)$ term a $\varphi(x/s)$ je formule.

Předchozí definice lze snadno rozšířit na definice substituce termu za term.

Substituce termu za proměnnou může vést k nechtěným situacím. Uvažujme např. formuli $\varphi = (\forall y)(y \leq x)$ která vyjadřuje, že x je větší než jakékoliv y. Substituce $y + y$ za x vede k $\varphi(x/y + y) = (\forall y)(y \leq y + y)$, což je jistě formule, která vyjadřuje něco jiného. Abychom zabránili takovým případům, definujeme tzv. korektní substituce.

Definice 3.16 Substituce termu t za proměnnou x ve formuli φ je ***korektní*** (t se nazývá ***substituovatelný*** za x v t), jestliže pro každou $y \in \mathrm{var}(t)$ platí, že žádná podformule formule φ, která je tvaru $(\forall y)\psi$, neobsahuje výskyt x, který je volným výskytem x ve φ.

Příklad 3.17 (Viz přednášky a cvičení)

Převod vět přirozeného jazyka na formule predikátové logiky

Příklad 3.18 Navrhněte formuli predikátové logiky, která vyjadřuje větu "Žádný plaz s výjimkou zmije není nebezpečný". Řešení: Větu přeformulujeme do podoby bližší predikátové logice: "Pro každé x platí: jestliže x je plaz a x není zmije, pak není pravda, že x je nebezpečný". Nyní navrhneme vhodný jazyk: unární relační symboly p, z, n. Vhodnou formulí je např. $(\forall x)((p(x) \wedge \neg z(x)) \Rightarrow \neg n(x))$.

Pozor: není to jednoznačný proces (více možných návrhů jazyka, více možných formulí při daném návrhu).

3.2 Sémantika predikátové logiky

Jazyk predikátové logiky obsahuje relační, funkční (a další) symboly. Ze těchto symbolů se skládají termy a formule. Samotné termy a formule nemají žádný význam, tj. term sám o sobě nemá žádnou hodnotu, formule sama o sobě nemá žádnou pravdivostní hodnotu. To je dobře patrné např. u termu $x + 0$: je do očí bijící, že k tomu, aby tento term měl hodnotu, musí mít nějakou hodnotu proměnná x (a dále, musíme interpretovat $+$ a 0, viz následující term $0 + 0$). Hůře je to patrné např. u termu $0 + 0$. Člověk má nutkání říci, že hodnotou tohoto termu je 0. To je ovšem hrubá chyba! Term $0 + 0$ je posloupností tří symbolů, nic víc; 0 je symbol konstanty (tj. 0-ární funkční symbol), $+$ je binární funkční symbol (přísně vzato, $0 + 0$ je přehlednějším způsobem zapsaný term $+(0, 0)$, to ale na celé věci nic nemění). Než řekneme, jaký prvek symbol 0 označuje a jakou binární funkci označuje symbol $+$ nemá smysl (nelze!) se ptát, jakou hodnotu má term $0 + 0$. Je naším záměrem, aby symboly mohly mít libovolnou (smysluplnou) interpretaci. Tím myslíme zhruba následující: Zvolíme vhodné univerzum M, tj. množinu prvků, které jsou pro naše úvahy relevantní (někdy se říká univerzum diskurzu, tj. univerzum rozpravy). V tomto univerzu přiřadíme významy relačním a funkčním symbolům daného jazyka $\mathcal{J}$ predikátové logiky, tj. pro každý relační symbol jazyka určíme konkrétní relaci v M, kterou tento relační symbol bude označovat, a pro každý funkční symbol jazyka určíme konkrétní funkci v M, kterou tento funkční symbol bude označovat. Takovou volbou univerza, relací a funkcí provedeme tzv. interpretaci jazyka (je tedy jasné, že jeden jazyk má mnoho možných interpretací); univerzum spolu s relacemi a funkcemi tvoří tzv. strukturu pro daný jazyk predikátové logiky (viz dále). Zvolíme-li interpretaci jazyka, zbývá v termech a formulích ještě jeden typ symbolů, které dosud nejsou interpretovány—proměnné. Interpretaci proměnných zprostředkovává tzv. ohodnocení, které každé proměnné jazyka přiřadí nějaký prvek univerza (je opět jasné, že při dané interpretaci jazyka existuje celá řada možných interpretací proměnných). Zvolíme-li interpretaci jazyka (tzv. strukturu pro daný jazyk) a interpretaci proměnných (tzv. ohodnocení proměnných pro daný jazyk v dané struktuře), můžeme se ptát po hodnotách termů (hodnotou termu je prvek univerza) a formulí (hodnotou formule je pravdivostní hodnota, tj. 0 nebo 1). Shrňme tedy a zdůrazněme: Termy i formule jsou pouhé řetězce symbolů bez jakéhokoli významu. Pojmy "hodnota termu" a “(pravdivostní) hodnota formule” nejsou smysluplné, smysluplné jsou pojmy “hodnota termu při dané interpretaci jazyka a dané interpretaci proměnných” a “(pravdivostní) hodnota formule při dané interpretaci jazyka a dané interpretaci proměnných” (neboli “hodnota termu v dané struktuře při daném ohodnocení” a “(pravdivostní) hodnota formule v dané struktuře při daném ohodnocení”).

Vraťme se k našemu příkladu, tj. k termům $0+0$ a $0+x$. Jsou to termy v jistém jazyce $\mathcal{J}$ (o kterém jsme zatím nemluvili). Jazykem $\mathcal{J}$ může být např. jazyk, kde $R = \{\leq\}$, $F = \{0, +\}$ a $\leq$ je binární, 0 je nulární (tj. symbol konstanty) a $+$ je binární. Zvolme jednu interpretaci jazyka $\mathcal{J}$: univerzem M nechť je množina $\mathbf{Z}$ všech celých čísel, binární relací v $\mathbf{Z}$, kterou označuje binární relační symbol $\leq$ nechť je obvyklá relace “menší nebo rovno” (označme ji $\leq^{\mathbf{M}}$), konstantou (tj. nulární funkcí) v $\mathbf{Z}$, kterou označuje nulární funkční symbol 0 nechť je číslo nula (označme ho $0^{\mathbf{M}}$), binární funkcí v $\mathbf{Z}$, kterou označuje binární funkční symbol $+$ nechť je obvyklé sčítání (označme ho $+^{\mathbf{M}}$). Zvolme dále interpretaci proměnných: nechť proměnným x a y jsou po řadě přiřazeny hodnoty pět a minus sto (a dalším proměnným hodnoty libovolné). Při takové interpretaci je hodnotou termu $0+0$ číslo nula ($0^{\mathbf{M}} +^{\mathbf{M}} 0^{\mathbf{M}}$), hodnotou termu $x+0$ je číslo pět. Formule $0 \leq x$ je při dané interpretaci pravdivá (neboť číslo, které je označeno symbolem 0, tj. číslo nula, je v relaci označené symbolem $\leq$, tj. v relaci “menší nebo rovno”, s číslem označeným symbolem x, tj. s číslem pět). Z podobného důvodu (zatím ovšem zdůvodňujeme pouze intuitivně) je pravdivá formule $(0 \leq x) \Rightarrow (y \leq y + x)$; ta je dokonce pravdivá při jakékoli interpretaci proměnných. Změníme-li interpretaci proměnných tak, že proměnné x bude přiřazeno číslo minus deset, bude formule $0 \leq x$ nepravdivá.

Uvedená interpretace jazyka však není jediná možná. Jinou interpretaci dostaneme, změníme-li naši původní interpretaci tak, že symbol 0 bude označovat číslo jedna. Při této interpretaci bude hodnotou termu $0+0$ číslo dvě. Zcela jinou interpretaci dostaneme, zvolíme-li za univerzum množinu všech čtvercových reálných matic, a označují-li symboly $\leq, 0, +$ po řadě relaci rovnosti matic, matici skládající se ze samých nul, a operaci násobení matic.

Definice 3.19 Nechť $\mathcal{J}$ je jazyk typu $\langle R, F, \sigma\rangle$. ***Struktura*** pro $\mathcal{J}$ je trojice $\mathbf{M} = \langle M, R^{\mathbf{M}}, F^{\mathbf{M}}\rangle$, která sestává z neprázdné množiny M, množiny $R^{\mathbf{M}} = \{r^{\mathbf{M}} \in L^{M^n} \mid r \in R, \sigma(r) = n\}$ relací a množiny $F^{\mathbf{M}} = \{f^{\mathbf{M}} : M^n \to M \mid f \in F, \sigma(r) = n\}$ funkcí. Jde-li o jazyk s rovností (tj. R obsahuje $\approx$), požadujeme, aby $\approx^{\mathbf{M}}$ byla ekvivalence na M.

Je-li $\mathbf{M}$ struktura pro jazyk $\mathcal{J}$ s rovností $\approx$ a je-li $\approx^{\mathbf{M}}$ (relace odpovídající $\approx$) identita na M (tj. nejen ekvivalence), nazývá se $\mathbf{M}$ strukturou se *striktní rovností*. Nebude-li uvedeno jinak, budeme pro jazyk s rovností uvažovat jen struktury se striktní rovností.

Poznámka 3.20 (1) Jinými slovy, struktura $\mathbf{M}$ pro jazyk $\mathcal{J}$ typu $\langle R, F, \sigma\rangle$ je systém relací a funkcí na jisté množině M; ke každému n-árnímu relačnímu symbolu $r \in R$ je ve struktuře odpovídající n-ární relace $r^{\mathbf{M}}$ v M, ke každému n-árnímu funkčnímu symbolu $f \in F$ je ve struktuře odpovídající n-ární funkce $f^{\mathbf{M}}$ v M.

(2) Nehrozí-li nebezpečí nedorozumění, vynecháváme někdy pro jednoduchost horní indexy a místo $r^{\mathbf{M}}$ a $f^{\mathbf{M}}$ píšeme jen r a f.

Příklad 3.21 (1) Uvažujme jazyk $\mathcal{J}$ z Příkladu 3.9 (1). Nechť M je množina všech lidí z daného regionu (např. z České republiky). Definujme relace $p^{\mathbf{M}}$, $d^{\mathbf{M}}$, $b^{\mathbf{M}}$, $s^{\mathbf{M}}$ následovně: $p^{\mathbf{M}}$, $d^{\mathbf{M}}$, $b^{\mathbf{M}}$ jsou unární relace, tj. podmnožiny M, definované

$$\begin{aligned}
p^{\mathbf{M}} &= \{m \in M \mid m \text{ má čistý příjem vyšší než 17 tis. Kč/měs.}\},\\
d^{\mathbf{M}} &= \{m \in M \mid m \text{ splácí pravidelně méně než 5 tis. Kč/měs.}\},\\
b^{\mathbf{M}} &= \{m \in M \mid m \text{ na živobytí zbývá více než 8 tis. Kč/měs.}\}
\end{aligned}$$

a $s^{\mathbf{M}}$ je binární relace

$$s^{\mathbf{M}} = \{\langle m_1, m_2\rangle \in M \times M \mid m_1 \text{ a } m_2 \text{ jsou manželé.}\}$$

Pak pro $R^{\mathbf{M}} = \{p^{\mathbf{M}}, d^{\mathbf{M}}, b^{\mathbf{M}}, s^{\mathbf{M}}\}$, $F^{\mathbf{M}} = \emptyset$ je $\mathbf{M} = \langle M, R^{\mathbf{M}}, F^{\mathbf{M}}\rangle$ strukturou pro jazyk $\mathcal{J}$.

(2) Jinou strukturu pro stejný jazyk dostaneme, pozměníme-li strukturu uvedenou v (1) tak, že

$$\begin{aligned}
p^{\mathbf{M}} &= \{m \in M \mid m \text{ má čistý příjem vyšší než 40 tis. Kč/měs.}\},\\
d^{\mathbf{M}} &= \{m \in M \mid m \text{ splácí pravidelně méně než 2 tis. Kč/měs.}\},\\
b^{\mathbf{M}} &= \{m \in M \mid m \text{ na živobytí zbývá více než 35 tis. Kč/měs.}\}
\end{aligned}$$

a $s^{\mathbf{M}}$ je binární relace

$$s^{\mathbf{M}} = \{\langle m_1, m_2\rangle \in M \times M \mid m_1 \text{ a } m_2 \text{ nejsou manželé.}\}$$

(3) Další strukturu pro stejný jazyk dostaneme, zvolíme-li $M = \{a, b, 1, 2\}$, $p^{\mathbf{M}} = \{a, b\}$, $d^{\mathbf{M}} = \{1, 2\}$, $b^{\mathbf{M}} = \emptyset$, $s^{\mathbf{M}} = \{\langle a, b\rangle, \langle 1, 2\rangle\}$.

(4) Uvažujme jazyk $\mathcal{J}$ z Příkladu 3.9 (2). Nechť $M = \{0, 1, -1, 2, -2, 3, -3, \dots\}$ je množina všech přirozených čísel. Definujme relace $p^{\mathbf{M}}$ (unární, tj. podmnožina M), $\leq^{\mathbf{M}}$ (binární) a funkce $c^{\mathbf{M}}$ (nulární, tj. vlastně prvek z M) a $\circ^{\mathbf{M}}$ (binární) následovně:

$$\begin{aligned}
p^{\mathbf{M}} &= \{m \in M \mid m \text{je větší nebo rovno nule}\},\\
\leq^{\mathbf{M}} &= \{\langle m_1, m_2\rangle \mid m_1 \text{ je menší nebo rovno } m_2\},\\
c^{\mathbf{M}} &= 0 \text{ (tj. } c^{\mathbf{M}} \text{ je číslo nula)},\\
m_1 \circ^{\mathbf{M}} m_2 &= m_1 + m_2 \text{ (tj. } \circ^{\mathbf{M}} \text{ je operace sčítání)}.
\end{aligned}$$

(5) Jinou strukturu pro jazyk $\mathcal{J}$ z Příkladu 3.9 (2) dostaneme, když změníme výše uvedenou strukturu tak, že $c^{\mathbf{M}} = 0$, popř. ještě definujeme $m_1 \circ^{\mathbf{M}} m_2 = m_1 \cdot m_2$ (násobení).

(6) Další strukturou pro jazyk $\mathcal{J}$ z Příkladu 3.9 (2) je struktura s nosičem

$$M = \{A \mid A \text{ je čtvercová matice řádu 10}\},$$

kde

$$\begin{aligned}
p^{\mathbf{M}} &= \{A \in M \mid A \text{ je regulární matice}\},\\
\leq^{\mathbf{M}} &= \{\langle A, B\rangle \mid a_{ij} \text{ je nejvýše rovno } b_{ij} \text{ pro všechna } 1 \leq i, j \leq n\},\\
c^{\mathbf{M}} &= I \text{ (tj. } c^{\mathbf{M}} \text{ jednotková matice)},\\
m_1 \circ^{\mathbf{M}} m_2 &= m_1 + m_2 \text{ (tj. } \circ^{\mathbf{M}} \text{ je operace sčítání matic)}.
\end{aligned}$$

(6) Další strukturou pro jazyk $\mathcal{J}$ z Příkladu 3.9 (2) je struktura s nosičem $M = \{a, b\}$, $p^{\mathbf{M}} = \{a, b\}$, $\leq^{\mathbf{M}} = \{\langle a, a\rangle, \langle b, b\rangle, \langle b, a\rangle\}$, $c^{\mathbf{M}} = b$ a operace $\circ^{\mathbf{M}}$ je dána tabulkou

$\circ^{\mathbf{M}}$	a	b
a	a	b
b	a	a

Jak tedy vidíme, k danému jazyku predikátové logiky existuje nekonečně mnoho struktur (variabilita je dána následujícím: nosič M může mít libovolný počet prvků, každý relační symbol r může být interpretován libovolnou relací $r^{\mathbf{M}}$ příslušné arity, každý funkční symbol f může být interpretován libovolnou funkcí $f^{\mathbf{M}}$ příslušné arity).

Nechť $\mathbf{M}$ je struktura pro jazyk $\mathcal{J}$. ***M-ohodnocení předmětových proměnných*** (krátce jen $\mathbf{M}$-ohodnocení, popř. jen ohodnocení) je zobrazení v přiřazující každé proměnné x prvek $v(x) \in M$. Jsou-li v a v' ohodnocení a x proměnná, píšeme $v =_x v'$ pokud pro každou proměnnou $y \neq x$ je $v(y) = v'(y)$, tj. v a v' se liší nejvýše v tom, jakou hodnotu přiřazují proměnné x.

Definice 3.22 Nechť v je $\mathbf{M}$-ohodnocení. ***Hodnota*** $\|t\|_{\mathbf{M},v}$ ***termu*** t při v je definována následovně:

(i) pro proměnnou x, $\|x\|_{\mathbf{M},v} = v(x)$;

(ii) pro $t = f(t_1, \ldots, t_n)$, $\|t\|_{\mathbf{M},v} = f^{\mathbf{M}}(\|t_1\|_{\mathbf{M},v}, \ldots, \|t_n\|_{\mathbf{M},v})$.

Poznámka 3.23 Uvědomme si, že podle Definice 3.22 je při daných $\mathbf{M}$ a v každému termu přiřazena právě jedna hodnota $\|t\|_{\mathbf{M},v}$. To lze nahlédnout použitím Věty 3.7, vezmeme-li za $\mathcal{V}$ vlastnost "hodnota $\|t\|_{\mathbf{M},v}$ je jednoznačně definována".

Lemma 3.24 *Hodnota $\|t\|_{\mathbf{M},v}$ nezávisí na hodnotách přiřazených ohodnocením v proměnným, které se v t nevyskytují, t.j. pro každá $\mathbf{M}$-ohodnocení v_1, v_2 splňující $v_1(x) = v_2(x)$ pro každé $x \in \mathrm{var}(t)$ je $\|t\|_{\mathbf{M},v_1} = \|t\|_{\mathbf{M},v_2}$.*

Důkaz. Strukturální indukcí. □

Poznámka 3.25 (jen na okraj) Podle Definice 3.22 a Lemma 3.24 indukuje každý term $t(x_1, \ldots, x_n)$ tzv. *termovou funkci* $t^{\mathbf{M}}$: pro $m_1, \ldots m_n \in M$ je

$$t^{\mathbf{M}}(m_1, \ldots, m_n) = \|t\|_{\mathbf{M},v},$$

kde v je $\mathbf{M}$-ohodnocení, pro které $v(x_1) = m_1, \ldots, v(x_n) = m_n$.

Příklad 3.26 term, $\|t\|_{\mathbf{M},v}$, termova funkce

Definice 3.27 ***Pravdivostní hodnota*** $\|\varphi\|_{\mathbf{M},v}$ ***formule*** φ ***při*** $\mathbf{M}$***-ohodnocení*** v je definována následovně:

(i) pro atomické formule:

$$\|r(t_1,\dots,t_n)\|_{\mathbf{M},v} = \begin{cases} 1 & \text{pokud } \langle \|t_1\|_{\mathbf{M},v},\dots,\|t_n\|_{\mathbf{M},v}\rangle \in r^{\mathbf{M}}, \\ 0 & \text{pokud } \langle \|t_1\|_{\mathbf{M},v},\dots,\|t_n\|_{\mathbf{M},v}\rangle \notin r^{\mathbf{M}}. \end{cases}$$

(ii) pro formule φ a ψ je

$$\|\neg\varphi\|_{\mathbf{M},v} = \begin{cases} 1 & \text{pokud } \|\varphi\|_{\mathbf{M},v} = 0, \\ 0 & \text{pokud } \|\varphi\|_{\mathbf{M},v} = 1. \end{cases}$$

$$\|\varphi \Rightarrow \psi\|_{\mathbf{M},v} = \begin{cases} 1 & \text{pokud } \|\varphi\|_{\mathbf{M},v} = 0 \text{ nebo } \|\psi\|_{\mathbf{M},v} = 1, \\ 0 & \text{jinak.} \end{cases}$$

(iii) pro formuli φ a proměnnou x je

$$\|(\forall x)\varphi\|_{\mathbf{M},v} = \begin{cases} 1 & \text{pokud pro každé } v' \text{ takové, že } v' =_x v, \text{ je } \|\varphi\|_{\mathbf{M},v'} = 1, \\ 0 & \text{jinak.} \end{cases}$$

Je-li $\|\varphi\|_{\mathbf{M},v} = 1$ ($\|\varphi\|_{\mathbf{M},v} = 0$), říkáme, že formule φ je pravdivá (nepravdivá) ve struktuře $\mathbf{M}$ při ohodnocení v. Uvědomte si, že říct "formule φ je pravdivá" nemá smysl; musíme říci, v jaké struktuře a při jakém ohodnocení! Že běžně říkáme např. "(formule) $(\forall x, y)x \le x + y$ je pravdivá", je způsobeno tím, že implicitně nějakou strukturu předpokládáme, většinou číselnou (např. celá čísla s běžnými relacemi a operacemi).

Poznámka 3.28 Tedy: $\|r(t_1,\dots,t_n)\|_{\mathbf{M},v} = 1$, právě když n-tice $\langle \|t_1\|_{\mathbf{M},v},\dots,\|t_n\|_{\mathbf{M},v}\rangle$ prvků $\|t_i\|_{\mathbf{M},v}$ z M patří do $r^{\mathbf{M}}$. Dále, význam spojek negace a implikace je stejný jako ve výrokové logice. K významu $\forall$: Za prvé, definice říká, že $(\forall x)\varphi$ je ve struktuře $\mathbf{M}$ při v pravdivá, právě když je ve struktuře $\mathbf{M}$ pravdivá formule φ při každém ohodnocení v', které všem proměnným různým od x přiřazuje stejné prvky jaké jim přiřazuje v. To je právě zamýšlený význam kvantifikátoru $\forall$. Za druhé, snadno se vidí, že

$$\|(\forall x)\varphi\|_{\mathbf{M},v} \quad = \quad \min\{\|\varphi\|_{\mathbf{M},v'} \mid v' =_x v\}.$$

(2) Snadno se přesvědčíme, že

$$\begin{aligned} \|\varphi \curlywedge \psi\|_{\mathbf{M},v} &= \|\varphi\|_{\mathbf{M},v} \wedge \|\psi\|_{\mathbf{M},v}, \\ \|\varphi \curlyvee \psi\|_{\mathbf{M},v} &= \|\varphi\|_{\mathbf{M},v} \vee \|\psi\|_{\mathbf{M},v}, \\ \|\varphi \Leftrightarrow \psi\|_{\mathbf{M},v} &= \|\varphi\|_{\mathbf{M},v} \leftrightarrow \|\psi\|_{\mathbf{M},v}, \end{aligned}$$

dále pak (ověřte!)

$$\|(\exists x)\varphi\|_{\mathbf{M},v} \quad = \quad \max\{\|\varphi\|_{\mathbf{M},v'} \mid v' =_x v\}.$$

Poznámka 3.29 Stejně jako u ohodnocení termů si uvědomme, že podle Definice 3.27 je při daných $\mathbf{M}$ a v každé formuli přiřazena právě jedna hodnota $\|\varphi\|_{\mathbf{M},v}$. To lze nahlédnout použitím Věty 3.8, vezmeme-li za $\mathcal{V}$ vlastnost "hodnota $\|\varphi\|_{\mathbf{M},v}$ je jednoznačně definována".

Lemma 3.30 *Hodnota $\|\varphi\|_{\mathbf{M},v}$ nezávisí na hodnotách přiřazených ohodnocením v proměnným, které se ve φ nevyskytují, t.j. pro každá $\mathbf{M}$-ohodnocení v_1, v_2 splňující $v_1(x) = v_2(x)$ pro každé $x \in \mathrm{var}(\varphi)$ je $\|\varphi\|_{\mathbf{M},v_1} = \|\varphi\|_{\mathbf{M},v_2}$.*

Důkaz. Strukturální indukcí. □

Příklad 3.31 Consider the **L**-structure **L**-structure **M** from Example **??** (1). Let v be an **M**-valuation such that $v(x) = 2$, $v(y) = 10$, $v(z) = 110$. Then for terms $t_1 = (x + y) + x$ and $t_2 = c + x$ we have $\|t_1\|_{\mathbf{M},v} = 14$, $\|t_2\|_{\mathbf{M},v} = 3$; for formulas $\varphi_1 = (x \leq y)$, $\varphi_2 = (y \leq x)$, $\varphi_3 = \mathrm{big}(x)$, $\varphi_4 = \mathrm{big}(z)$, $\varphi_5 = (\forall x)(\forall y)((\mathrm{big}(x) \otimes (x \leq y)) \Rightarrow \mathrm{big}(y))$, $\varphi_6 = (\forall x)(\mathrm{big}(x + c) \Rightarrow \mathrm{big}(x))$ we have $\|\varphi_1\|_{\mathbf{M},v} = 1$, $\|\varphi_2\|_{\mathbf{M},v} = 0$, $\|\varphi_3\|_{\mathbf{M},v} = 0$, $\|\varphi_4\|_{\mathbf{M},v} = 0.1$, $\|\varphi_5\|_{\mathbf{M},v} = 1$, $\|\varphi_6\|_{\mathbf{M},v} = 0.99$. Note that the truth values of φ_5 and φ_6 do not depend on v (Lemma 3.30).

Podle Definice 3.27 a Lemma 3.30 indukuje každá formule $\varphi(x_1, \ldots, x_n)$ (tj. formule s volnými proměnnými mezi $x_1, \ldots, x_n$) n-ární relaci $\varphi^{\mathbf{M}}$ v M: pro $m_1, \ldots m_n \in M$ je

$$\langle m_1, \ldots, m_n \rangle \in \varphi^{\mathbf{M}} \text{ právě když } \|\varphi\|_{\mathbf{M},v} = 1, \qquad (8)$$

kde v je $\mathbf{M}$-ohodnocení, pro které $v(x_1) = m_1, \ldots, v(x_n) = m_n$.

Příklad 3.32 formule, $\|\varphi\|_{\mathbf{M},v}$, indukovana relace

Definice 3.33 Formule φ se nazývá ***tautologie ve struktuře*** (pravdivá ve struktuře) **M**, jestliže $\|\varphi\|_{\mathbf{M},v} = 1$ pro každé **M**-ohodnocení v. Formule φ se nazývá ***tautologie*** **M** (vždy pravdivá), jestliže je to tautologie v každé struktuře **M**.

φ je tedy tautologie, jestliže pro libovolnou strukturu **M** a libovolné ohodnocení v je $\|\varphi\|_{\mathbf{M},v} = 1$.

Definice 3.34 ***Teorie*** v jazyce $\mathcal{J}$ predikátové logiky je libovolná množina T formulí tohoto jazyka. Struktura **M** jazyka $\mathcal{J}$ se nazývá ***model teorie*** T (píšeme $\mathbf{M} \models T$, popř. $\|T\|_{\mathbf{M}} = 1$), jestliže každá formule z T je pravdivá v **M**.

Poznámka 3.35 Pojem teorie je zcela přirozený. Běžně se říká "S tvojí teorií nesouhlasím" apod. Přitom teorií rozumíme soubor tvrzení, které daná osoba zastává. Soubor tvrzení v predikátové logice představuje množina formulí.

Příklad 3.36 Uvažujme jazyk $\mathcal{J}$ typu $\langle R, F, \sigma \rangle$, kde $R = \{r\}$, $F = \emptyset$ a $\sigma(r) = 2$. Struktury pro $\mathcal{J}$ jsou $\mathbf{M} = \langle M, \{r^{\mathbf{M}}\}, \emptyset \rangle$, kde $r^{\mathbf{M}}$ je binární relace na M (tj. struktury jsou vlastně binární relace na M). Struktura $\mathbf{M}$ je modelem teorie $T = \{(\forall x)r(x,x), (\forall x,y)(r(x,y) \Rightarrow r(y,x))\}$, právě když je relace $r^{\mathbf{M}}$ reflexivní a symetrická.

Definice 3.37 Množina S formulí ***sémanticky plyne*** z množiny T formulí (píšeme $T \models S$; píšeme také $T \models \varphi$, jestliže $S = \{\varphi\}$, podobně když $T = \{\varphi\}$), jestliže každý model T je modelem S.

Tedy $T \models S$, právě když v každé struktuře, ve které jsou pravdivé všechny formule z T, jsou také pravdivé všechny formule z S.

Příklad 3.38 Formule

$$\psi = (\forall x,y,z,w)((r(x,y) \curlywedge r(y,z) \curlywedge r(z,w)) \Rightarrow r(x,w))$$

sémanticky plyne z formule

$$\varphi = (\forall x,y,z)((r(x,y) \curlywedge r(y,z)) \Rightarrow r(x,z)),$$

tj. $\varphi \models \psi$. Obrácené vyplývání, tj. $\psi \models \varphi$, neplatí. Viz přednáška.

3.3 Axiomatický systém predikátové logiky

Axiomy jsou formule následujících tvarů: formule tvaru (1)–(3) a dále

$$(\forall x)\varphi \Rightarrow \varphi(x/t) \tag{9}$$

je-li t substituovatelný za x

$$(\forall x)(\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow (\forall x)\psi) \tag{10}$$

nemá-li x ve φ volný výskyt.

(9) se nazývá axiom substituce, (10) se nazývá axiom distribuce.
V případě logiky s rovností máme ještě *axiomy rovnosti*:

$$x \approx x \tag{11}$$

$$x \approx y \Rightarrow y \approx x \tag{12}$$

$$x \approx y \curlywedge y \approx z \Rightarrow x \approx z \tag{13}$$

$$x_1 \approx y_1 \curlywedge \cdots \curlywedge x_n \approx y_n \Rightarrow f(x_1, \ldots, x_n) \approx f(y_1, \approx, y_n) \tag{14}$$

pro každý n-ární $f \in F$

$$x_1 \approx y_1 \curlywedge \cdots \curlywedge x_n \approx y_n \curlywedge r(x_1, \ldots, x_n) \Rightarrow r(y_1, \approx, y_n) \tag{15}$$

pro každý n-ární $r \in R$.

Odvozovací pravidla jsou *modus ponens* (viz výše) a pravidlo *generalizace* (G, zobecnění), které říká

$$\text{z } \varphi \text{ odvoď } (\forall x)\varphi.$$

Poznámka 3.39 Všimněme si, že všechny axiomy jsou tautologie (snadno se vidí, ale později přesto dokázat). Dále si všimněme, že omezení u axiomů (9) a (10) jsou podstatná (tj. bez nich by se nejednalo o tautologie). Skutečně, je-li φ formule $\neg\ (\forall y)r(x, y)$, t proměnná y, pak (10) je formule $(\forall) \neg\ (\forall y)r(x, y) \Rightarrow \neg\ (\forall y)r(y, y)$, která není tautologií (např. není splněna v žádné struktuře $\mathbf{M}$ s aspoň dvěma prvky, ve které $r^{\mathbf{M}}$ je reflexivní relace). Dále, jsou-li obě φ i ψ formulí $r(x)$, je (10) formulí $(\forall x)(r(x) \Rightarrow r(x)) \Rightarrow (r(x) \Rightarrow (\forall x)r(x))$, což není tautologie (např. není splněna ve struktuře $\mathbf{M}$ s aspoň dvěma prvky, ve které je $r^{\mathbf{M}}$ jednoprvková množina).

Definice 3.40 ***Důkaz*** formule φ z množiny T formulí je libovolná posloupnost $\varphi_1, \dots, \varphi_n$, pro kterou platí, že $\varphi_n = \varphi$ a každá φ_i

- je axiomem
- nebo je formulí z T
- nebo plyne z předchozích formulí důkazu pomocí odvozovacího pravidla MP (tj. existují $j, k < i$ $(1 \leq j \leq n,\ 1 \leq k \leq n)$ tak že φ_k je formule $\varphi_j \Rightarrow \varphi_i$) nebo odvozovacího pravidla G (tj. existuje $j < i$ tak, že φ_i je formulí $(\forall x)\varphi_j$).

Formule se nazývá ***dokazatelná*** z T, existuje-li důkaz této formule z T (zapisujeme $T \vdash \varphi$, popř. jen $\vdash \varphi$, je-li T prázdná množina).

Poznámka 3.41 Stejně jako ve výrokové logice máme dva pojmy vyplývání formule z množiny formulí: sémantické vyplývání $(T \models \varphi)$ a syntaktické vyplývání $(T \vdash \varphi)$. Máme také dva pojmy platnosti formule: $\models \varphi$ označuje platnost φ v sémantickém smyslu (pravdivost), $\vdash \varphi$ označuje platnost φ v syntaktickém smyslu (dokazatelnost).

Formule se nazývá *výrokově dokazatelná* z T, existuje-li její důkaz z T, ve kterém se vyskutují pouze axiomy typů (1)–(3) a odvozovací pravidlo MP. T se nazývá *výrokově sporná*, jestliže každá formule je z T výrokově dokazatelná. Formule φ a ψ se nazývají dokazatelně ekvivalentní z T, je-li $T \vdash \varphi \Leftrightarrow \psi$.

Lemma 3.42 *Nahradíme-li v tautologii výrokové logiky výrokové symboly libovolnými formulemi predikátové logiky, dostaneme formuli predikátové logiky, která je výrokově dokazatelná.*

Důkaz. (podrobněji přednášky) Je-li φ ona tautologie, pak dle věty o úplnosti pro výrokovou logiku je dokazatelná. Nahradíme-li v jejím důkazu ve výrokové logice výrokové symboly zmíněnými formulemi predikátové logiky, dostaneme důkaz v predikátové logice, prokazující, že výsledná formule je výrokově dokazatelná. $\square$

Příklad 3.43 Protože $p \Rightarrow p$ a $p \Rightarrow (\neg\, p \Rightarrow q)$ jsou tautologie výrokové logiky jsou formule $(\forall x)x \leq y \Rightarrow (\forall x)x \leq y$ a $y \leq x + y \Rightarrow (\neg\, y \leq x + y \Rightarrow x \approx 0)$ (výrokově) dokazatelné formule predikátové logiky.

Věta 3.44 (o dedukci) *Pro formuli φ bez volných proměnných a množinu T formulí platí*

$$T, \varphi \vdash \psi \textit{ právě když } T \vdash \varphi \Rightarrow \psi.$$

Důkaz. Analogicky jako pro výrokovou logiku, viz přednášky. □

Příklad 3.45 Ukažme, že požadavek, aby φ neměla volné proměnné, je podstatný. [řešení: Nechť φ je $r(x)$ (r je unární relační symbol), ψ nechť je $(\forall x)r(x)$, T nechť je prázdná množina; pak užitím pravidla generalizace dostaneme $T, \varphi \vdash \psi$, ovšem $T \vdash \varphi \Rightarrow \psi$ neplatí. Kdyby to platilo, pak by dle věty o korektnosti (kterou uvedeme za chvíli) bylo $T \models \varphi \Rightarrow \psi$, tj. $r(x) \Rightarrow (\forall x)r(x)$ by byla tautologií, což neplatí (např. neplatí ve struktuře s množinou přirozených čísel jako univerzem, kde r je interpretován jako množina čísel větších než 5 (nebo jakákoli množina různá od množiny všech přirozených čísel)).]

3.3.1 Další tvrzení, která jsou analogiemi tvrzení z výrokové logiky

Věta o dedukci platná pro predikátovou logiku (Věta 3.44) je typickým příkladem řady tvrzení, která mají z našeho pohledu (znalců výrokové logiky) následující charakter. Jsou analogiemi nám známých tvrzení z výrokové logiky. Liší se však od nich jednak tím, že v nich tvrdíme něco o jiných objektech (např. tvrzení se týkají formulí predikátové logiky, nikoli formulí výrokové logiky), jednak tím, že obsahují dodatečné předpoklady, které jsou v případě výrokové logiky zbytečné. Jde zejména o následující: věta o důkazu sporem, věta o důkazu rozborem případů, věta o neutrální formuli, věta o ekvivalenci. Dobrým cvičením je zformulovat si uvedené věty pro případ predikátové logiky a dokázat je (analogicky jako ve výrokové logice, jen je třeba dát pozor na některé předpoklady, tak jako tomu bylo u věty o dedukci).

3.3.2 Tvrzení o formulích s kvantifikátory

Další řadou tvrzení o predikátové logice jsou tvrzení o kvantifikátorech. Tato tvrzení pochopitelně nemají analogie ve výrokové logice. Začneme následujícím užitečným tvrzením.

Věta 3.46 (o uzávěru) *Pro každou teorii T a formuli φ platí*

$$T \vdash \varphi \quad \textit{právě když} \quad T \vdash (\forall x)\varphi.$$

Tedy, formule je dokazatelná, právě když je dokazatelný její libovolný uzávěr.

Důkaz. Předpokládejme $T \vdash \varphi$. Pak $T \vdash (\forall x)\varphi$ díky G (podrobněji, je-li $\dots, \varphi$ důkaz φ z T, je $\dots, \varphi, (\forall x)\varphi$ důkazem $(\forall x)\varphi$ z T).

Předpokládejme $T \vdash (\forall x)\varphi$. Protože $(\forall x)\varphi \Rightarrow \varphi$ je axiom (9), kde $t = x$, je $\vdash (\forall x)\varphi \Rightarrow \varphi$, tedy $T \vdash \varphi$ použitím MP. □

Věta 3.47 *Pro formule φ, ψ platí*

$$\vdash (\forall x)(\varphi \Rightarrow \psi) \Rightarrow ((\forall x)\varphi \Rightarrow (\forall x)\psi) \tag{16}$$

$$\vdash (\forall x)(\varphi \Rightarrow \psi) \Rightarrow ((\exists x)\varphi \Rightarrow (\exists x)\psi) \tag{17}$$

$$\vdash (\forall x)(\varphi \Rightarrow \psi) \Leftrightarrow (\varphi \Rightarrow (\forall x)\psi) \tag{18}$$

není-li x volná ve φ

$$\vdash (\forall x)(\varphi \Rightarrow \psi) \Leftrightarrow ((\exists x)\varphi \Rightarrow \psi) \tag{19}$$

není-li x volná v ψ

$$\vdash (\exists x)(\varphi \Rightarrow \psi) \Leftrightarrow (\varphi \Rightarrow (\exists x)\psi) \tag{20}$$

není-li x volná ve φ

$$\vdash (\exists x)(\varphi \Rightarrow \psi) \Leftrightarrow ((\forall x)\varphi \Rightarrow \psi) \tag{21}$$

není-li x volná v ψ

$$\vdash (\forall x)(\forall y)\varphi \Leftrightarrow (\forall y)(\forall x)\varphi \tag{22}$$

$$\vdash (\exists x)(\exists y)\varphi \Leftrightarrow (\exists y)(\exists x)\varphi \tag{23}$$

$$\vdash (\exists x)(\forall y)\varphi \Leftrightarrow (\forall y)(\exists x)\varphi \tag{24}$$

Důkaz. □

Další tvrzení — viz přednášky. Ke zkoušce je třeba znát principy práce s kvantifikátory a umět provádět jednoduché úvahy o kvantifikátorech v tom rozsahu, jak bylo probíráno zejm. na cvičení (o distribuci kvantifikátorů, o záměně pořadí kvantifikátorů a implikace, kvantifikace a negace, o záměně pořadí kvantifikátorů).

Na přednáškách jsme uvedli další tvrzení, uvádějící dokazatelné formule predikátové logiky: o distribuci kvantifikátoru, o rovnosti, o záměně pořadí kvantifikátorů a implikace, o záměně pořadí kvantifikátorů.

3.4 Úplnost predikátové logiky

Cílem je dostat se k větě o úplnosti (tj. "dokazatelné = vždy pravdivé") pro predikátovou logiku; nejdříve uvedeme větu o korektnosti. Úplnost poté ukážeme metodou tzv. henkinovských rozšíření teorií.

Věta 3.48 (o korektnosti) *Pro libovolnou teorii T a libovolnou formuli φ jazyka teorie T platí, že*

$$z\ T \vdash \varphi \text{ plyne } T \models \varphi.$$

Důkaz. Na stejném principu jako pro výrokovou logiku (podrobněji viz přednášky). □

Poznámka 3.49 Jednoduchým důsledkem je fakt: sporná teorie nemá model. Totiž, kdyby byla T sporná, pak pro každou formuli φ by platilo $T \vdash \varphi$ i $T \vdash\neg\, \varphi$. Dle Věty o korektnosti by muselo být v každém modelu teorie T pravdivé φ i $\neg\, \varphi$, což není možné.

Přistoupíme nyní k důkazu věty o úplnosti. Nejprve uvedeme několik pomocných tvrzení (sama o sobě jsou však zajímavá).

Definice 3.50 Teorie S je ***rozšířením*** teorie T, jestliže jazyk S obsahuje jazyk T a je-li každý axiom teorie T dokazatelný v S. Rozšíření S teorie T se nazývá ***konzervativní*** , jestliže každá formule jazyka teorie T, která je dokazatelná v S, je dokazatelná také v T. Teorie jsou ***ekvivalentní*** , jestliže jsou jedna druhé rozšířením.

Poznámka 3.51 Snadno se ukáže, že je-li S rozšířením T, je každá formule dokazatelná v T dokazatelná také v S.

Věta 3.52 (o konstantách) *Je-li S rozšíření T takové, že jazyk S obsahuje nové konstanty $c_1, \ldots, c_n$, které jsou od sebe různé, ale S neobsahuje nové axiomy. Pak pro každou formuli φ jazyka teorie T platí*

$$T \vdash \varphi \text{ právě když } S \vdash \varphi(x_1/c_1, \ldots, x_n/c_n).$$

Důkaz. Tvrzení dokážeme nejdřív pro jednu konstantu (tj. pro $n = 1$); pak ho indukcí rozšíříme na více konstant. Předpokládejme tedy $n = 1$ a pro jednoduchost pišme c místo c_1.

"$\Rightarrow$": Předpokládejme $T \vdash \varphi$. Nechť $\ldots, \varphi$ je důkaz φ z T. V teorii S lze tento důkaz prodloužit o formule $(\forall x)\varphi$ (G na φ), $(\forall x)\varphi \Rightarrow \varphi(x/c)$ (axiom (9)), $\varphi(x/c)$ (MP na předchozí formule). Tedy

$$\ldots, \varphi, (\forall x)\varphi, (\forall x)\varphi \Rightarrow \varphi(x/c), \varphi(x/c)$$

je důkaz z S, a proto $S \vdash \varphi$.

"$\Leftarrow$": Nechť $S \vdash \varphi(x/c)$, tj. nechť existuje důkaz $\varphi_1, \ldots, \varphi_k$ formule $\varphi(x/c)$ z S (tj. $\varphi_k = \varphi(x/c)$). Vezměme proměnnou y, která se v tomto důkazu nevyskytuje (tj. nevyskytuje se v žádné z φ_i). Indukcí ukážeme, že $\varphi_1(c/y), \ldots, \varphi_k(c/y)$ je důkaz z T.

Předpokládejme tedy (indukční předpoklad), že $\varphi_1(c/y), \ldots, \varphi_{i-1}(c/y)$ je důkaz z T a ukažme, že i $\varphi_1(c/y), \ldots, \varphi_k(c/y)$ je důkaz z T. Protože $\varphi_1, \ldots, \varphi_k$ je důkaz z S, mohou nastat následující případy.

- φ_i je některý z axiomů (**??**–**??**): Snadno se vidí, že pak je $\varphi_i(c/y)$ také axiomem. Je ovšem třeba ověřit, že v případě axiomů (**??**)–(**??**) jsou splněny předpoklady (to se snadno ukáže; ověřte).

- φ_i je formule z S: Podle předpokladů jsou formule z S a z T stejné (REFORMULATE), tedy φ_i je i formulí z T. Potom ale φ_i neobsahuje konstantu c, a tedy φ_i je totožná s $\varphi_i(c/y)$.
- φ_i vyplývá z předchozích formulí pomocí odvozovacího pravidla MP nebo G: Pokud φ_i vyplývá z φ_j a φ_k $(j, k < i)$ pomocí MP (tj. např. $\varphi_j = \varphi_k \Rightarrow \varphi_i$), pak $\varphi_j(c/y)$ je $\varphi_k(c/y) \Rightarrow \varphi_i(c/y)$ a tedy $\varphi_i(c/y)$ vyplývá pomocí MP z $\varphi_j(c/y)$ je $\varphi_k(c/y)$. Pokud φ_i vyplývá z φ_j $(j < i)$ pomocí G (tj. φ_i je $(\forall x)\varphi_j$), pak $\varphi_i(c/y)$, což je $[(\forall x)\varphi_j](c/y)$, je rovna $(\forall x)[\varphi_j(c/y)]$, neboť podle předpokladu je y různá od x, a tedy $\varphi_i(c/y)$ vyplývá z $\varphi_j(c/y)$ pomocí G.

Ukázali jsme tedy $T \vdash [\varphi(x/c)](c/y)$. Protože $[\varphi(x/c)](c/y)$ je formule $\varphi(x/y)$, máme $T \vdash \varphi(x/y)$. Je vidět, že x je substituovatelná za y ve formuli $\varphi(x/y)$. Proto platí, že prodloužíme-li důkaz $\dots, \varphi(x/y)$ z T na posloupnost

$$\dots, \varphi(x/y), (\forall y)\varphi(x/y), (\forall y)\varphi(x/y) \Rightarrow [\varphi(x/y)](y/x), [\varphi(x/y)](y/x),$$

dostali jsme důkaz formule $[\varphi(x/y)](y/x)$ z T (postupně použitím G, axiomu (**??**), MP). Uvědomíme-li si, že $[\varphi(x/y)](y/x)$ je formulí φ, vidíme $T \vdash \varphi$.

Nyní tvrzení snadno rozšíříme na $c_1, \dots, c_n$: Zatím jsme dokázali, že za uvedených podmínek platí $T \vdash \varphi$ právě když $S \vdash \varphi(x/c)$. Označíme-li T' teorii S (tj. její jazyk obsahuje c_1) a jako S' teorii, která vznikne z S přidáním c_2 do jazyka, dostaneme použitím dokázaného tvrzení na formuli $\psi = \varphi(x_1/c_1)$, že $T' \vdash \psi$, právě když $S' \vdash \psi(x_2/c_2)$, tj. máme $S \vdash \varphi(x_1/c_1)$, právě když $S' \vdash \varphi(x_1/c_1, x_2/c_2)$. Zároveň máme $T \vdash \varphi$, právě když $S \vdash \varphi(x_1/c_1)$, tedy celkem $T \vdash \varphi$, právě když $S' \vdash \varphi(x_1/c_1, x_2/c_2)$. Opakovaným použitím této úvahy nakonec dostaneme požadované tvrzení. □

Poznámka 3.53 Speciálně tedy je za podmínek Věty 3.52 S konzervativní rozšíření T.

Definice 3.54 Formule φ je ***variantou*** formule ψ, jestliže existuje posloupnost $\varphi = \theta_1, \theta_2, \dots, \theta_n = \psi$ formulí tak, že pro každé $i < n$ vznikne formule θ_{i+1} z formule θ_i nahrazením jedné podformule formule θ_i, která je tvaru $(\forall x)\chi$ (popř. $(\exists x)\chi$) formulí $(\forall y)\chi(x/y)$ (popř. $(\exists y)\chi(x/y)$), kde proměnná y je substituovatelná za x v χ a není volná v χ.

Věta 3.55 (o variantách) *Je-li ψ varianta φ, pak $\vdash \varphi \Leftrightarrow \psi$.*

Důkaz. Viz přednášky. □

Následuje užitečné (a zajímavé) tvrzení.

Lemma 3.56 *Pro teorii T, formuli φ a libovolný uzávěr $\overline{\varphi}$ formule φ je*

$$T \vdash \varphi \quad \textit{právě když} \quad T, \neg\, \overline{\varphi} \textit{ je sporná.}$$

Důkaz. Nechť $T \vdash \varphi$. Dle Věty 3.46 je $T \vdash \overline{\varphi}$. Nechť ψ je libovolná formule. Dokážeme $T, \neg\,\overline{\varphi} \vdash \psi$. Máme

$\vdash \overline{\varphi} \Rightarrow (\neg\,\overline{\varphi} \Rightarrow \psi)$ [použití Lemma 3.42 na tautologii $p \Rightarrow (\neg\, p \Rightarrow q)$ výrokové logiky]

$T \vdash \neg\,\overline{\varphi} \Rightarrow \psi$ [MP]

$T, \neg\,\overline{\varphi} \vdash \psi$ [MP],

což znamená, že $T, \neg\,\overline{\varphi}$ je sporná.

Naopak, nechť je $T, \neg\,\overline{\varphi}$ sporná. Pak máme

$T, \neg\,\overline{\varphi} \vdash \overline{\varphi}$ [ze spornosti $T, \neg\,\overline{\varphi}$]

$T \vdash \neg\,\overline{\varphi} \Rightarrow \overline{\varphi}$ [věta o dedukci]

$\vdash (\neg\,\overline{\varphi} \Rightarrow \varphi) \Rightarrow \overline{\varphi}$ [použití Lemma 3.42 na tautologii $(\neg\, p \Rightarrow p) \Rightarrow p$ výrokové logiky]

$T \vdash \overline{\varphi}$ [MP]

$T \vdash \varphi$ [věta o uzávěru (Věta 3.46)]. □

Definice 3.57 Teorie T se nazývá ***henkinovská*** [3], jestliže pro každou formuli $\varphi(x)$ se v jazyce teorie T vyskytuje konstanta c tak, že formule $(\exists x)\varphi \Rightarrow \varphi(x/c)$ je dokazatelná v T. Konstanta c se nazývá ***henkinovská konstanta***, $(\exists x)\varphi \Rightarrow \varphi(x/c)$ se nazývá ***henkinovská formule*** příslušná formuli φ.

Věta 3.58 (o henkinovské konstantě) *Je-li $\varphi(x)$ formule jazyka teorie T a je-li S rozšíření T vzniklé přidáním (henkinovské) konstanty c_φ a (henkinovské) formule $(\exists x)\varphi \Rightarrow \varphi(x/c)$, pak S je konzervativním rozšířením teorie T.*

Důkaz. Označme R teorii vzniklou z T přidáním c_φ (tj. S vznikne přidáním $(\exists x)\varphi \Rightarrow \varphi(x/c)$ k R). Nechť pro formuli ψ jazyka teorie T platí $S \vdash \psi$, tj. $R, (\exists x)\varphi \Rightarrow \varphi(x/c) \vdash \psi$. Abychom prokázali konzervativnost rozšíření S, musíme dokázat $T \vdash \psi$. Zvolme proměnnou y, která se nevyskytuje v žádné z φ a ψ. Podle Věty o dedukci máme

$$R \vdash [(\exists x)\varphi \Rightarrow \varphi(x/c_\varphi)] \Rightarrow \psi$$

a podle Věty o konstantách (uvážíme-li, že $\{[(\exists x)\varphi \Rightarrow \varphi(x/y)] \Rightarrow \psi\}(y/c_\varphi)$ je $[(\exists x)\varphi \Rightarrow \varphi(x/c_\varphi)] \Rightarrow \psi)$ dále

$$T \vdash [(\exists x)\varphi \Rightarrow \varphi(x/y)] \Rightarrow \psi,$$

z čehož použitím G dostaneme

$$(\forall y)[(\exists x)\varphi \Rightarrow \varphi(x/y)] \Rightarrow \psi,$$

odtud dále (podle pravidel práce s kvantifikátory)

$$(\exists y)[(\exists x)\varphi \Rightarrow \varphi(x/y)] \Rightarrow \psi,$$

[3] L. Henkin, ...

odtud dále (podle pravidel práce s kvantifikátory)

$$[(\exists x)\varphi \Rightarrow (\exists y)\varphi(x/y)] \Rightarrow \psi,$$

odkud dostaneme

$$T \vdash \psi$$

použitím MP na předcházející dokazatelnost. Totiž, platí $\vdash (\exists x)\varphi \Rightarrow (\exists x)\varphi$ a dle Věty o variantách je tedy $\vdash (\exists x)\varphi \Rightarrow (\exists y)\varphi(x/y)$ (můžeme tedy aplikovat MP). □

Věta 3.59 (o henkinovském rozšíření) *Ke každé teorii existuje henkinovská teorie, která je jejím konzervativním rozšířením.*

Důkaz. Nechť T_0 je výchozí teorie. K té sestrojíme henkinovské rozšíření následovně. Nejprve sestrojíme posloupnost teorií $T_1, T_2, \ldots$ následovně.

Konstrukce $T_{i+1} z T_i$: Jazykem T_{i+1} bude jazyk T_i obohacený o henkinovské konstanty všech formulí jazyka T_i s jednou volnou proměnnou (naším cílem je totiž "odstranit nehenkinovskost" T_i); axiomy T_{i+1} jsou axiomy T_i a henkinovské axiomy příslušné ke všem formulím jazyka T_i s jednou volnou proměnnou.

Tvrdíme, že každá T_{i+1} je konzervativním rozšířením T_i. Musíme tedy ukázat, že je-li ψ formule jazyka T_i, pro kterou $T_{i+1} \vdash \psi$, pak $T_i \vdash \psi$. Nechť je tedy $T_{i+1} \vdash \psi$ a nechť $\psi_1, \ldots, \psi_n$ je příslušný důkaz. Uvažujme všechny konstanty $c_{\varphi_1}, \ldots, c_{\varphi_k}$, které se vyskytují v důkazu $\psi_1, \ldots, \psi_n$, ale nepatří do jazyka T_i. Uvažujme dále teorie $S_0, S_1, \ldots, S_k$ takové, že $S_0 = T_i$, S_{i+1} vznikne z S_i rozšířením o $c_{\varphi_{i+1}}$ a příslušný henkinovský axiom. Pak je $\psi_1, \ldots, \psi_n$ je důkazem ψ z S_k, a tedy podle Věty o henkinovské konstantě je $S_{k-1} \vdash \psi$, z čehož postupnou aplikací Věty o henkinovské konstantě dostaneme $S_{k-2} \vdash \psi$, $\ldots$, $S_0 \vdash \psi$, tj. $T_i \vdash \psi$.

Protože žádná z T_i ještě nemusí být henkinovská (rozšířením jazyka totiž mohou vzniknout nové formule, které porušují henkinovskost), uvažujme dále $T = \bigcup_{i=1,2,\ldots} T_i$. Přímo z konstrukce T plyne, že je to henkinovská teorie (ověřte). Navíc je konzervativním rozšířením původní T_0, neboť je-li ψ nějaká formule jazyka T_0 a $\varphi_1, \ldots, \varphi_n$ je důkaz ψ z T, pak je to také důkaz ψ z nějakého T_i (pro dostatečně velké i), a tedy z konzervativnosti T_i plyne, že ψ je dokazatelná z T_0. □

Definice 3.60 Teorie T se nazývá ***úplná*** , jestliže je berzesporná a jestliže pro každou uzavřenou formuli φ je buď $T \vdash \varphi$, nebo $T \vdash \neg \varphi$.

Věta 3.61 (o zúplňování teorií) *Ke každé bezesporné teorii existuje její rozšíření se stejným jazykem, které je úplnou teorií.*

Důkaz. Předpokládejme pro jednoduchost, že množina všech uzavřených formulí daného jazyka je spočetná (což například platí, je spočetný daný jazyk;

není-li množina všech formulí spočetná, můžeme ji tzv. dobře uspořádat a postupovat odpovídajícím principem indukce), tj. všechny uzavřené formule můžeme uspořádat do posloupnosti $\varphi_1, \varphi_2, \ldots$. Nechť T je daná bezesporná teorie. Pro $i = 1, 2, \ldots$ budeme sestrojovat teorie $T_i \supseteq T$, které budou bezespornými rozšířeními teorie T. Položme navíc $T_0 = T$.

Konstrukce pro dané i: Předpokládejme, že pro $j < i$ máme sestrojeny $T_j \supseteq T$, které jsou bezespornými rozšířeními teorie T. Označme

$$S = \bigcup_{j<i} T_j.$$

Platí, že S je bezesporným rozšířením T. Skutečně, kdyby byla S sporná, existoval by z S důkaz $\psi_1, \ldots, \psi_n$ nějaké vždy nepravdivé formule φ. Pak ale existuje $j' < i$ tak, že veškeré předpoklady z S, které jsou prvky důkazu $\psi_1, \ldots, \psi_n$, patří do $T_{j'}$, tedy $\psi_1, \ldots, \psi_n$ je důkazem z $T_{j'}$, což není možné, protože podle předpokladu je $T_{j'}$ bezesporná.

Je-li $S \cup \{\varphi_i\}$ bezesporná, položme $T_i = S \cup \{\varphi_i\}$. V tom případě je T_i bezesporné rozšíření T.

Je-li $S \cup \{\varphi_i\}$ sporná, položme $T_i = S \cup \{\neg\, \varphi_i\}$. Protože je $S \cup \{\varphi_i\}$ sporná, je podle Lemma 3.56 $S \vdash \neg\, \varphi_i$ (při aplikaci Lemma 3.56 si uvědomte, že φ_i je uzavřená a že $S \cup \{\varphi_i\}$ je sporná, právě když je sporná $S \cup \{\neg\neg\, \varphi_i\}$). Protože je S bezesporná, je bezesporná i $T_i = S \cup \{\neg\, \varphi_i\}$.

Hledaným rozšířením T' je pak $T' = \bigcup_{i=1,2,\ldots} T_i$. Bezespornost T' se ukáže podobně jako bezespornost S výše; úplnost T' je zřejmá z jeho konstrukce. $\Box$

Definice 3.62 (kanonická struktura) ***Kanonická struktura*** $\mathbf{M}_T$ teorie T je dána následovně:

- univerzem $\mathbf{M}_T$ je množina všech uzavřených termů (tj. neobsahujících proměnné) jazyka T;
- pro n-ární relační symbol $r \in R$ je relace $r^{\mathbf{M}_T}$ definována předpisem
$$\langle t_1, \ldots, t_n \rangle \in r^{\mathbf{M}_T} \text{ právě když } T \vdash r(t_1, \ldots, t_n);$$
- pro n-ární funkční symbol $f \in F$ je funkce $f^{\mathbf{M}_T}$ definována předpisem
$$f^{\mathbf{M}_T}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n).$$

Poznámka 3.63 Všimněme si, že v definici kanonické struktury je použit elegantní trik: K dané teorii je definována struktura, tj. sémantický pojem, která je však sestavena jen ze syntaktických prvků a pojmů. Nosičem kanonické struktury je množina všech uzavřených termů, tedy termů, které neobsahují proměnné. Aby tedy kanonická struktura vůbec existovala, je nutné, aby jazyk dané teorie obsahoval symboly konstant. Funkce $f^{\mathbf{M}_T}$ jsou definovány jednoduše: pro prvky $t_1, \ldots, t_n$ univerza (jsou to uzavřené termy) je výsledkem aplikace definované funkce $f^{\mathbf{M}_T}$ na $t_1, \ldots, t_n$ řetězec $f(t_1, \ldots, t_n)$, což je uzavřený term, tedy prvek univerza. Konstrukce relací $r^{\mathbf{M}_T}$ je důmyslnější. Využívá se při něm (netriviální) pojem dokazatelnosti. n-tice $\langle t_1, \ldots, t_n \rangle$ je v relaci $r^{\mathbf{M}_T}$, právě když je formule $r(t_1, \ldots, t_n)$ dokazatelná v teorii T.

Věta 3.64 (o kanonické struktuře) *Je-li T úplná henkinovská teorie, pak $\mathbf{M}_T$ je modelem T.*

Důkaz. Dokážeme nejdříve následující tvrzení (označne ho $(*)$): pro každou uzavřenou instanci φ' formule φ je $T \vdash \varphi'$, právě když $\|\varphi'\|_{\mathbf{M}_T} = 1$ (uzavřená instance formule φ je každá taková formule φ', která vznikne z φ aplikací nějaké korektní substituce, tj. některé proměnné se nahradí termy, přitom φ' je sama uzavřenou formulí; $'$ tedy označuje nějaké korektní nahrazení proměnných termy, které z φ udělá φ'). Z toho speciálně plyne, že pro každou uzavřenou formuli φ je $T \vdash \varphi$, právě když $\|\varphi\|_{\mathbf{M}_T} = 1$ (neboť uzavřená formule je uzavřenou instancí sama sebe).

Dále můžeme předpokládat, že každá formule z T je uzavřená. Skutečně, je-li θ uzávěrem ψ, je $S \cup \{\psi\} \vdash \varphi$, právě když $S \cup \{\theta\} \vdash \varphi$ (dokáže se podobnou úvahou jako Věta o uzávěru). Tedy T dokazuje stejné formule jako teorie, která z T vznikne nahrazením formulí s volnými proměnnými jejich uzávěry.

Z toho pak plyne, že $\mathbf{M}_T$ je modelem T následovně. Je-li φ uzavřená formule z T, pak je $T \vdash \varphi$, a tedy $\|\varphi\|_{\mathbf{M}_T} = 1$ (φ je pravdivá v $\mathbf{M}_T$) podle $(*)$.

Zbývá dokázat tvrzení $(*)$: Tvrzení dokážeme strukturální indukcí přes φ.

Je-li φ atomická, tj. φ je tvaru $r(t_1, \ldots, t_n)$, pak tvrzení plyne přímo z definice $r^{\mathbf{M}_T}$ (a z toho, že $\|\varphi\|_{\mathbf{M}_T} = 1$ znamená $\langle t_1, \ldots, t_n \rangle \in r^{\mathbf{M}_T}$).

Předpokládejme, že tvrzení platí pro φ a ψ a dokažme, že platí také pro $\neg \varphi$, $\varphi \Rightarrow \psi$ a $(\exists x)\varphi$.

$\neg \varphi$: Máme $(\neg \varphi)' = \neg (\varphi')$, tedy $T \vdash \neg (\varphi')$ právě když (z bezespornosti a úplnosti T) není $T \vdash \varphi'$, což podle předpokladu platí právě když $\|\varphi'\|_{\mathbf{M}_T} = 0$, což je právě když $\| \neg (\varphi')\|_{\mathbf{M}_T} = 1$.

$\varphi \Rightarrow \psi$: Uvědomme si následující: (a) $(\varphi \Rightarrow \psi)' = \varphi' \Rightarrow \psi'$; (b) $\|\varphi' \Rightarrow \psi'\|_{\mathbf{M}_T} = 1$ právě když $\|\varphi'\|_{\mathbf{M}_T} = 1$ implikuje $\|\psi'\|_{\mathbf{M}_T} = 1$; (c) z $T \vdash \varphi' \Rightarrow \psi'$ plyne, že $T \vdash \varphi'$ implikuje $T \vdash \psi'$. Z toho plyne (promyslete!), že stačí ověřit, že neplatí-li $T \vdash \varphi' \Rightarrow \psi'$, pak neplatí, že $T \vdash \varphi'$ implikuje $T \vdash \psi'$. To ale platí. Totiž, neplatí-li $T \vdash \varphi' \Rightarrow \psi'$, pak z úplnosti T plyne $T \vdash \neg (\varphi' \Rightarrow \psi')$, tedy $T \vdash \varphi' \wedge \neg \psi'$, tedy $T \vdash \varphi'$ a $T \vdash \neg \psi'$. Z bezespornosti a úplnosti T plyne, že neplatí $T \vdash \psi'$.

$(\exists x)\varphi$: Je-li $T \vdash (\exists x)\varphi$, pak z henkinovskosti teorie T dostaneme použitím MP, že $T \vdash \varphi(x/c)$. Z předpokladu, že tvrzení platí pro φ ($\varphi(x/c)$ je uzavřenou instancí φ), dostaneme $\|\varphi(x/c)\|_{\mathbf{M}_T} = 1$, tedy z definice je $\|(\exists x)\varphi\|_{\mathbf{M}_T} = 1$. Naopak, není-li $T \vdash (\exists x)\varphi$, pak z úplnosti T je $T \vdash \neg (\exists x)\varphi$, tedy $T \vdash (\forall x) \neg \varphi$. Pro každý uzavřený term t máme dále $\vdash (\forall x) \neg \varphi \Rightarrow \neg \varphi(x/t)$ (neboť dotčená dokazatelná formule je axiom (9)), použitím MP tedy dostaneme $T \vdash \neg \varphi(x/t)$, a protože tvrzení platí pro $\neg \varphi$, dostáváme $\| \neg \varphi(x/t)\|_{\mathbf{M}_T} = 1$. Protože univerzum M_T obsahuje pouze uzavřené termy, platí pro každé ohodnocení e (uvědomme si, že jedinou volnou proměnnou formule $\neg \varphi$ je x), že $\| \neg \varphi\|_{\mathbf{M}_T,e} = 1$, tedy $\|(\forall x) \neg \varphi\|_{\mathbf{M}_T,e} = 1$, a tedy $\|(\exists x)\varphi\|_{\mathbf{M}_T,e} = 0$, což bylo třeba dokázat. Důkaz $(*)$ je hotov. □

Definice 3.65 (kanonická struktura s rovností) $\mathbf{M}_T/\approx^{\mathbf{M}_T}$ — viz přednášky

Věta 3.66 (o kanonické struktuře s rovností) *Je-li T úplná henkinovská teorie s rovností, pak $\mathbf{M}_T/\approx^{\mathbf{M}_T}$ je modelem T.*

Důkaz. Nebude u zkoušky požadován. □

Poznámka 3.67 Nechť teorie S je rozšířením teorie T. Mají-li S a T stejný jazyk, je zřejmě každý model teorie S také modelem teorie T. Je-li jazyk teorie S bohatší než jazyk teorie T, tj. $R_T \subset R_S$ nebo $F_T \subset F_S$, kde R_T, F_T a R_S, F_S označují po řadě relační a funkční symboly jazyka teorie T a S, je z každého modelu $\mathbf{M}_S$ teorie S možné vytvořit model $\mathbf{M}_T$ teorie T vypuštěním příslušných relací a funkcí, tj. $M_T = M_S$, $R^{\mathbf{M}_T} = \{r^{\mathbf{M}_S} \mid r \in R_T\}$, $F^{\mathbf{M}_T} = \{f^{\mathbf{M}_S} \mid f \in F_T\}$. Zjednodušeně však můžeme i v tomto případě říkat, že každý model teorie S je také modelem teorie T.

Věta 3.68 (o úplnosti) *(1) Každá bezesporná teorie má model. (2) Pro každou teorii T a každou formuli φ platí, že*

$$\textit{je-li } T \models \varphi, \ \textit{pak } T \vdash \varphi.$$

Důkaz. (1) Nechť T je bezesporná teorie. Dle Věty 3.59 existuje její henkinovské rozšíření T', které je jejím konzervativním rozšířením. Protože T' je konzervativní rozšíření, plyne z bezespornosti T, že T' je také bezesporná (kdyby byla T' sporná, platilo by pro jakoukoli formuli φ jazyka teorie T, že $T' \vdash \varphi$ i $T' \vdash \neg \varphi$. Z konzervativnosti by dále plynulo, že $T \vdash \varphi$ i $T \vdash \neg \varphi$, a tedy T by byla sporná). Dle Věty 3.61 existuje rozšíření T'' teorie T', které má stejný jazyk jako teorie T' a je úplnou teorií. Protože je T' henkinovská teorie, je henkinovská i T''. Dle Věty 3.64, popř. Věty 3.66 (pokud se jedná o jazyk s rovností), existuje model teorie T''. Ten je však také modelem teorie T (viz Poznámku 3.67).

(2) Označme $\overline{\varphi}$ libovolný uzávěr formule φ. Kdyby neplatilo $T \vdash \varphi$, pak by teorie $T, \neg \overline{\varphi}$ byla dle Lemma 3.56 bezesporná. Podle (1) by tedy $T, \neg \overline{\varphi}$ měla model $\mathbf{M}$. V $\mathbf{M}$ je pravdivá $\neg \overline{\varphi}$, tedy je v něm nepravdivá $\overline{\varphi}$. Protože ve struktuře je formule pravdivá, právě když je v ní pravdivý její uzávěr, je v $\mathbf{M}$ nepravdivá formule φ. $\mathbf{M}$ je tedy modelem teorie T, ve kterém neplatí φ, což je spor s předpokladem $T \models \varphi$. □

Teorie T je množina formulí. Podteorie S dané teorie je její podmnožina (tj. $S \subseteq T$).

Věta 3.69 (o kompaktnosti) *Teorie má model, právě když každá její konečná podteorie má model.*

3.5 Prenexní tvar, dokazatelnost v predikátové logice vs. výroková dokazatelnost

Toto nebylo v přednáškách (!), u zkoušky nebude obsah sekce 3.5 požadován.

Následující výsledky jsou mimo jiné významné pro logické základy tzv. automatizovaného dokazování. Prenexní tvar jej jistou normální formou formule, Hilbert-Ackermannova a Herbrandova věta převádějí pro určité případy dokazatelnost (složitější pojem) na výrokovou dokazatelnost (jednodušší pojem).

Teorie se nazývá otevřená, jsou-li všechny její axiomy otevřené formule. Instance dané formule je formule, která z dané formule vznikne korektní substitucí termů za proměnné.

Věta 3.70 (Hilbert-Ackermannova) *Otevřená teorie T je sporná, právě když existuje konečná teorie S se stejným jazykem jako T, která má za axiomy pouze instance axiomů rovnosti a instance axiomů teorie T a která je výrokově sporná.*

Důkaz. Nebude u zkoušky požadován. □

Definice 3.71 Formule je v ***prenexním tvaru*** , je-li ve tvaru $(Q_1x_1)\dots(Q_nx_n)\varphi$, kde Q_i je buď $\forall$ nebo $\exists$, x_i jsou navzájem různé proměnné a formule φ je otevřená (tj. neobsahuje kvantifikátory) ; $(Q_1x_1)\dots(Q_nx_n)$ se nazývá prefix, φ jádro.

Příklad 3.72 Formule $(\forall x)(\forall y)(\exists z)(\forall u)(r(x,y,u) \leq f(y))$ je v prenexním tvaru; formule $(\forall x)(\exists y)(r(y) \vee (\forall x)s(y))$ ani $(\forall x)(\forall y)(\exists x)r(x,y)$ nejsou.

Věta 3.73 (o prenexním tvaru) *Ke každé formuli φ existuje formule ψ v prenexním tvaru tak, že $\vdash \varphi \Leftrightarrow \psi$.*

Důkaz. Pro každou formuli $\varphi = \varphi_0$ sestrojíme pomocí postupných úprav posloupnost vzájemně ekvivalentních formulí $\varphi_0, \varphi_1, \dots, \varphi_n$ tak, φ_i je ekvivalentní s φ_{i+1} a φ_n je v prenexním tvaru. Úpravy jsou takové, že převádějí kvantifikátory na začátek formule. Snadno se vidí, že lze uvažovat jen následující případy:

Není-li φ_i v prenexním tvaru, obsahuje podformuli tvaru $\neg\ (Qx)\chi$ nebo $(Qx)\chi \Rightarrow \psi$ nebo $\chi \Rightarrow (Qx)\psi$ (Q je některý kvantifikátor). Tuto podformuli nahradíme ve φ_i s ní ekvivalentní formulí a dostaneme φ_{i+1}. Nechť k Q je Q' ten druhý kvantifikátor (tj. $\forall'$ je $\exists$ a $\exists'$ je $\forall$).

Podformuli tvaru $\neg\ (Qx)\chi$ nahradíme $(Q'x)\neg\ \chi$.

U podformule tvaru $(Qx)\chi \Rightarrow \psi$ (popř. $\chi \Rightarrow (Qx)\psi$) nejprve přejmenujeme proměnné tak, aby $(Qx)\chi$ a ψ (popř. χ a $(Qx)\psi$) neobsahovaly společné proměnné. Přejmenováním získaná formule $(Qy)\chi' \Rightarrow \psi'$ (popř. $\chi' \Rightarrow (Qy)\psi'$) je ekvivalentní s $(Qx)\chi \Rightarrow \psi$ (popř. $\chi \Rightarrow (Qx)\psi$). Formule $(Qy)\chi' \Rightarrow \psi'$ (popř. $\chi' \Rightarrow (Qy)\psi'$) je dále dle větu o záměně pořadí kvantifikátoru a implikace ekvivalentní formuli $(Q'y)(\chi' \Rightarrow \psi')$ (popř. $(Qy)(\chi' \Rightarrow \psi')$). Podformuli $(Qx)\chi \Rightarrow \psi$ nahradíme formulí $(Q'y)(\chi' \Rightarrow \psi')$, podformuli $\chi \Rightarrow (Qx)\psi$ nahradíme $(Qy)(\chi' \Rightarrow \psi')$. □

Důkaz předchozí věty má konstruktivní charakter:

Příklad 3.74 Převedte formuli $(\forall y)[(\forall z)(\exists x)r(x,y) \Rightarrow (\forall x) \neg (\forall y)s(x,y)]$ do prenexního tvaru. Nejprve přesuneme negaci a získáme $(\forall y)[(\forall z)(\exists x)r(x,y) \Rightarrow (\forall x)(\exists y) \neg s(x,y)]$. V podformuli $[(\forall z)(\exists x)r(x,y) \Rightarrow (\forall x)(\exists y) \neg s(x,y)]$ provedeme přeznačení dostaneme $[(\forall z)(\exists x)r(x,y) \Rightarrow (\forall x')(\exists y') \neg s(x',y')]$. Provedením úprav v této formuli dostaneme s ní ekvivalentní formuli $(\exists z)(\forall x)(\forall x')(\exists x')[r(x,y) \Rightarrow \neg s(x',y')]$. Celkem je tedy původní formule ekvivalentní formuli $(\forall)(\exists z)(\forall x)(\forall x')(\exists x')[r(x,y) \Rightarrow \neg s(x',y')]$, která již je v prenexním tvaru.

Nechť φ je uzavřená formule v prenexním tvaru. Indukcí budeme definovat formuli φ^S (tzv. *skolemovskou variantu*formule φ); ta bude stále v prenexní formě, nebude obsahovat existenční kvantifikátory, ale bude možná obsahovat nové funkční symboly (půjde tedy o formuli nad rozšířeným jazykem). Neobsahuje-li φ existenční kvantifikátory, je φ^S přímo formule φ. Předpokládejme, že φ obsahuje existenční kvantifikátory, že je ve tvaru $(\forall x_1)\ldots(\forall x_n)(\exists y)\psi$ a že χ^S je ji6 definováno pro každou formuli χ, která má méně existenčních kvantifikátorů než φ. Formulí φ^S bude formule $[(\forall x_1)\ldots(\forall x_n)\psi(y/f(x_1,\ldots,x_n))]^S$, kde f je nový n-ární funkční symbol.

Formule φ^S je tedy tvaru $(\forall x_1 \ldots x_n)\psi$, kde ψ neobsahuje kvantifikátory. ψ pak nazýváme *otevřenou skolemovskou variantou* formule φ a značíme φ^{OS}.

Příklad 3.75 Sestrojte skolemovskou variantu formule $(\forall x)(\exists y)(\forall z)(\forall u)(\exists v)(\forall w)[r(x,y) \vee (g(z,v) \leq h(u,w))]$. [řešení: je to např. formule $(\forall x,z,u,w)[r(x,f_1(x)) \vee (g(z,f_2(x,z,u)) \leq h(u,w))]$, kde f_1, f_2 jsou nové funkční symboly.]

Věta 3.76 (Herbrandova) *Pro uzavřenou formuli φ v prenexním tvaru je $\vdash \varphi$, právě když existuje konečná výrokově sporná teorie T, která obsahuje pouze instance formule $(\neg\, \varphi)^{OS}$ a instance axiomů rovnosti.*

Důkaz. Nebude u zkoušky požadován. □

4 Úvod do fuzzy logiky

Proč potřebujeme fuzzy logiku? Co je základní motivací.

- Klasická logika nestačí při modelování tzv. vágních tvrzení, např. "Petr je velký.", "Teplota je vysoká". Uvedená tvrzení často intuitivně považujeme za ani úplně nepravdivá, ani úplně pravdivá, tj. za tvrzení, jejichž pravdivostní hodnota leží mezi 0 a 1, např. je to 0.9 (skoro pravda), 0.5 napůl pravda, 0.1 (skoro nepravda).
- S vágními tvrzeními se setkáme téměř při každém popisu reálného světa. Jde tedy o netriviální a širokou oblast.
- Jako první se uvedenou problematikou z pohledu možných aplikací zabýval Lotfi A. Zadeh (dnes stále aktivní jako profesor na University of California v Berkley) v nesmírně vlivné práci "Fuzzy sets. *Information and Control* (1965)".

- Fuzzy logika je dnes bohatě rozvinutá jak po stránce komerčně úspěšných aplikací (zejm. fuzzy regulátory), tak po stránce teoretických základů.

Problém volby struktur pravdivostních hodnot, základní požadavky.

- Množinu pravdivostních hodnot budeme značit L. Přirozeně požadujeme, aby $0, 1 \in L$ (0 označuje "(úplná) nepravda", 1 označuje "(úplná) pravda"). Požadujeme, aby L byla částečně uspořádána relací $\leq$.

- Příklady. $L = [0, 1]$; $L = \{0, 1\}$ (klasická logika); L je konečný řetězec (např. podmnožina $[0, 1]$); $L = \{0, 1\} \times \{0, 1\}$ s uspořádáním po složkách (pak $\langle a, b\rangle$ reprezentuje např. názor dvou expertů, první říká a, druhý b, to je přirozený příklad nelineárně uspořádané struktury pravdivostních hodnot).

- Další požadavky: Musí existovat operace na L modelující logické spojky (zejm. $\otimes$ pro konjunkci, $\rightarrow$ pro implikaci,). Tyto operace by měly mít přirozené vlastnosti odpovídající vlastnostem požadovaným po logických spojkách.

- Ukažme si, jak lze tímto způsobem dojít k jedné ze základních struktur pravdivostních hodnot ve fuzzy logice, k tzv. reziduovaným svazům. Začněme požadavky, které by měla splňovat operace $\otimes$. Je-li $\&$ symbol spojky konjunkce, pak přirozeně chceme, aby se pravdivostní hodnota $\|\varphi \,\&\, \psi\|$ konjunkce formulí φ a ψ dala vypočítat pomocí operace $\otimes$ z pravdivostní hodnoty $\|\varphi\|$ formule φ a pravdivostní hodnoty $\|\psi\|$ formule ψ. Chceme tedy mít $\|\varphi \,\&\, \psi\| = \|\varphi\| \otimes \|\psi\|$. Dále je přirozené požadovat, aby $\otimes$ byla komutativní (tj. aby platilo $a \otimes b = b \otimes a$; chceme totiž, aby pravdivostní hodnota formule $\varphi \,\&\, \psi$ byla stejná jako pravdivostní hodnota formule $\psi \,\&\, \varphi$, označíme-li $a = \|\varphi\|$, $b = \|\psi\|$, pak vlastně chceme $a \otimes b = \|\varphi\| \otimes \|\psi\| = \|\varphi \,\&\, \psi\| = \|\psi \,\&\, \varphi\| = \|\psi\| \otimes \|\varphi\| = b \otimes a$, tedy chceme komutativitu $\otimes$). Z podobného důvodu chceme, aby $\otimes$ byla asociativní. Dalším požadavkem je $a \otimes 1 = a$ (neboť chceme, aby pro "úplně" pravdivou formuli τ, tj. $\|\tau\| = 1$, a libovolnou formuli φ platilo $\|\varphi \,\&\, \tau\| = \|\varphi\|$). Operace $\otimes$ by dále měla být monotónní, tj. mělo by platit, že z $a_1 \leq a_2$ a $b_1 \leq b_2$ plyne $a_1 \otimes b_1 \leq a_2 \otimes b_2$ (chceme totiž, aby pravdivostní hodnota konjunkce rostla s pravdivostními hodnotami konjungovaných formulí). Další požadavek, který však nerozvedeme do detailů, se týká vztahu operací $\otimes$ a $\rightarrow$. Lze ukázat, že chceme-li, aby i ve fuzzy logice "dobře fungovalo" pravidlo *modus ponens*, vede to na požadavek, aby platilo, že $a \otimes b \leq c$ právě když $a \leq b \rightarrow c$. Posledním požadavkem je, aby v L existovala vzhledem k uspořádání $\leq$ infima i suprema libovolných podmnožin (tj. aby existovala $\bigwedge K$ a $\bigvee K$ pro libovolnou $K \subseteq L$). Tento požadavek plyne z toho, že chceme přirozeným způsobem definovat pravdivostní hodnoty formulí $(\forall x)\varphi$ a $(\exists x)\varphi$. Ukažme to na jednoduchém příkladě: Mějme tvrzení φ_i $(i \in I)$, každé z nich nechť má pravdivostní hodnotu $\|\varphi_i\|$. Co by mělo být pravdivostní hodnotou tvrzení "pro každé $i \in I$ platí φ_i"? Přirozený argument říká, že by to měla být nejmenší ze všech $\|\varphi_i\|$ a pokud tato neexistuje, pak by to mělo být infimum $\bigwedge_{i\in I} \|\varphi_i\|$. Podobně dojdeme od tvrzení "existuje $i \in I$ tak, že platí φ_i" k požadavku existence suprem.

Reziduované svazy a t-normy Výše uvedené požadavky na strukturu pravdivostních hodnot vedou k následující definici.

Definice 4.1 ***Úplný reziduovaný svaz*** je struktura $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$, kde

(1) $\langle L, \wedge, \vee, 0, 1\rangle$ je úplný svaz (s nejmenším prvkem 0 a největším prvkem 1),

(2) $\langle L, \otimes, 1\rangle$ je komutativní monoid (tj. $\otimes$ je binární operace na L, která je komutativní, asociativní a platí $a \otimes 1 = a$),

(3) $\otimes, \rightarrow$ jsou binární operace na L (nazývané "násobení" a "reziduum") splňující $a \otimes b \leq c$ právě když $a \leq b \rightarrow c$ (tzv. podmínka adjunkce).

Všimněme si, že úplné reziduované svazy vyhovují výše formulovaným požadavkům. Každý požadavek se zde objevuje jako jedna z podmínek, které musí úplný reziduovaný svaz splňovat. To však neplatí beze zbytku: např. se v definici reziduovaného svazu neobjevuje požadavek, aby $\otimes$ byla monotónní. Monotónnost $\otimes$ v reziduovaném svazu vždy platí (lze to dokázat z ostatních podmínek).

Uvedeme příklady úplných reziduovaných svazů, které se nejčastěji používají.

Mezi nejčastěji používané struktury pravdivostních hodnot patří ty, které mají za nosič reálný interval $[0,1]$ s přirozeným uspořádáním, tedy $a \wedge b = \min(a,b)$, $a \vee b = \max(a,b)$. Na nich se používají tři páry adjungovaných operací $\otimes$ a $\rightarrow$: Łukasiewiczovy operace ($a \otimes b = \max(a+b-1, 0)$, $a \rightarrow b = \min(1-a+b, 1)$), Gödelovy operace ($a \otimes b = \min(a,b)$, $a \rightarrow b = 1$ pro $a \leq b$ a $= b$ jinak), součinové operace ($a \otimes b = a \cdot b$, $a \rightarrow b = 1$ pro $a \leq b$ a $= b/a$ jinak).

Další důležitou množinou pravdivostních hodnot je $\{a_0 = 0, a_1, \ldots, a_n = 1\}$ ($a_0 < \cdots < a_n$) se dvěma páry adjungovaných operací: $a_k \otimes a_l = a_{\max(k+l-n,0)}$ a $a_k \rightarrow a_l = a_{\min(n-k+l,n)}$ (první); $a_k \otimes a_l = a_{\min(k,l)}$ a $a_k \rightarrow a_l = a_n$ pro $a_k \leq a_l$ a $a_k \rightarrow a_l = a_l$ jinak (druhá, ta vznikne restrikcemi Gödelových operací na množinu $\{a_0 = 0, a_1, \ldots, a_n = 1\}$).

Je-li $L = \{0,1\}$, $\otimes$ je operace klasické konjunkce a $\rightarrow$ je operace klasické implikace, pak příslušný reziduovaný svaz (ve které je uspořádání dáno vztahem $0 \leq 1$) je svazem pravdivostních hodnot klasické logiky (a je to až na jiným způsobem definované operace dvouprvková Booleova algebra). Obecněji platí, že Booleovy algebry jsou "vlastně" reziduované svazy. Přesněji, připomeňmě, že Booleova algebra je (resp. bývá tak nejčastěji definována) svaz s 0 a 1 (0 je nejmenší a 1 největší prvek svazu), který je distributivní (tj. platí $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$) a komplementární (tj. existuje unární operace $'$ zvaná komplementace splňující $a \wedge a' = 0$ a $a \vee a' = 1$). Nyní platí, že je-li $\mathbf{B} = \langle B, \wedge, \vee, ', 0, 1\rangle$ Booleova algebra, pak definijeme-li $L := B$, $a \otimes b := a \wedge b$, $a \rightarrow b := a' \vee b$, je $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ reziduovaný svaz splňující $a \otimes b = a \wedge b$ a $a'' = a$. Naopak, je-li $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ reziduovaný svaz splňující $a \otimes b = a \wedge b$ a $a'' = a$, pak definujeme-li $B := L$ a $a' := a \rightarrow 0$, je $\mathbf{B} = \langle B, \wedge, \vee, ', 0, 1\rangle$ Booleova algebra. Vidíme tedy, že reziduované svazy jsou bohaté struktury zahrnující např. Booleovy algebry (ale také další významné algebraické struktury jako např. Heytingovy algebry, MV-algebry, algebry tzv. lineární logiky a další).

Definice 4.2 ***t-norma*** je operace $\otimes : [0,1] \times [0,1] \rightarrow [0,1]$, která je komutativní, asociativní, monotónní a splňující $x \otimes 1 = 1$. t-norma se nazývá *zleva spojitá* (*spojitá*), jsou-li obě funkce $a \otimes \cdot$ a $\cdot \otimes a$ zleva spojité (spojité).

Výše uvedené operace $\otimes$ (Łukasiewicz, Gödel (minimum) a součin) jsou t-normy, dokonce spojité (ověřte!).

Věta 4.3 $\langle [0,1], \min, \max, \otimes, \rightarrow, 0, 1\rangle$ *je úplný reziduovaný svaz, právě když* $\otimes$ *je zleva spojitá t-norma a* $\rightarrow$ *je dáno vztahem*

$$a \rightarrow b = \sup\{c \mid a \otimes c \leq b\}.$$

Ověřte, že uvedený vztah mezi $\rightarrow$ a $\otimes$ v případě Łukasiewicz, Gödel (minimum) a product (součin) platí.

Tak například výše uvedené Łukasiewicz, Gödel (minimum) a product (součin) jsou spojité t-normy. Platí, že každou spojitou t-normu lze ze tří výše uvedených získat jednoduchou konstrukcí.

V reziduovaném svazu definujmeme některé odvozené operace. Mezi nejdůležitější patří tzv. bireziduum ($\leftrightarrow$) a negace ($\neg$) definované následovně:

$$a \leftrightarrow b = (a \rightarrow b) \wedge (b \rightarrow a)$$

a

$$\neg a = a \rightarrow 0.$$

Odvoďte formule pro $\leftrightarrow$ a $\neg$ pro případy Łukasiewicz, Gödel a součin.

Výroková fuzzy logika: dvě pojetí syntaxe, zejména ale neohodnocená syntaxe; sémantika. Existuje mnoho tzv. fuzzy logik. Každá fuzzy logika je dána nějakou třídou $\mathcal{L}$ struktur pravdivostních hodnot. Třída $\mathcal{L}$ je dána nějakými dodatečnými požadavky kladenými na logické spojky (resp. operace na L). Tak například chceme-li, aby platilo $\neg\neg a = a$ (tzv. zákon dvojí negace), omezíme se na třídu $\mathcal{L}$ pravdivostních hodnot definovanou $\mathcal{L} = \{\mathbf{L} \mid \mathbf{L}$ je úplný reziduovaný svaz splňující $\neg\neg a = a\}$. Dalším požadavkem může být, aby L byla lineárně uspořádaná apod. Z výše uvedeného je patrné, že požadujeme-li $\neg\neg a = a$ a $a \otimes b = a \wedge b$, pak $\mathcal{L}$ sestává právě z úplných Booleových algeber. Požadujeme-li navíc, aby množiny pravdivostních hodnot byly lineárně uspořádané, obsahuje $\mathcal{L}$ jedinou strukturu pravdivostních hodnot—dvouprvkovou Booleovu algebru, tj. strukturu pravdivostních hodnot klasické logiky.

V dalším se budeme zabývat základními pojetími fuzzy logiky.

Neohodnocená syntaxe. Nechť $\mathcal{L}$ je tedy nějaká třída struktur pravdivostních hodnot, např. $\mathcal{L}$ je třída všech reziduovaných svazů na $[0,1]$ se spojitou t-normou $\otimes$. Jazyk fuzzy logiky s neohodnocenou syntaxí obsahuje na rozdíl od jazyka klasické výrokové logiky tyto symboly spojek: $⊗$, $\Rightarrow$, $⩓$, $⩔$, dále symboly některých pravdivostních hodnot, např. $\mathsf{0}$, $\mathsf{1}$ (a můžeme chtít a pro každé $a \in [0,1]$). Je-li $\mathbf{L} \in \mathcal{L}$ struktura pravdivostních hodnot, pak $\mathbf{L}$-ohodnocení e je libovolné zobrazení z množiny všech výrokových symbolů do množiny L všech pravdivostních hodnot, tj. pro výrokový symbol p je $e(p) \in L$ chápána jako pravdivostní hodnota tvrzení, které je označeno p.

Formule jsou definovány jako obvykle (každý výrokový symbol je formule; $\mathsf{0}$ a $\mathsf{1}$ (a obecně a) jsou formule; jsou-li φ, ψ formule, pak $(\varphi ⊗ \psi)$, $(\varphi \Rightarrow \psi)$, $(\varphi ⩓ \psi)$, $(\varphi ⩔ \psi)$ jsou formule).

Pravdivostní hodnota formule φ při ohodnocení e se definuje následovně: $\|p\|_e = e(p)$; $\|\mathsf{0}\|_{=}0$ a $\|\mathsf{1}\|_e = 1$ (a obecně $\|\mathsf{a}\|_e = a$) jsou formule; $\|\varphi ⊗ \psi\|_e = \|\varphi\|_e \otimes \|\psi\|_e$, $\|\varphi \Rightarrow \psi\|_e = \|\varphi\|_e \rightarrow \|\psi\|_e$, $\|\varphi ⩓ \psi\|_e = \|\varphi\|_e \wedge \|\psi\|_e$, $\|\varphi ⩔ \psi\|_e = \|\varphi\|_e \vee \|\psi\|_e$.

Formule φ se nazývá: $\mathbf{L}$-tautologie, pokud $\|\varphi\|_e = 1$ pro každé $\mathbf{L}$-ohodnocení e; $\mathcal{L}$-tautologie (popř. pouze tautologie, pokud je $\mathcal{L}$ zřejmá z kontextu), pokud je to $\mathbf{L}$-tautologie pro každou $\mathbf{L} \in \mathcal{L}$.

Příklad 4.4 Přesvědčte se, že je-li $\mathcal{L}$ jednoprvková, jejímž jediným prvkem $\mathbf{L}$ je dvouprvková Booleova algebra, pak výše uvedené pojmy se shodují s odpovídajícími pojmy z klasické logiky (speciálně: pojem pravdivostní hodnota formule; tautologie v klasické logice jsou právě $\mathcal{L}$-tautologie). Tedy klasická logika "vznikne" z obecné fuzzy logiky vhodnou parametrizací (volbou $\mathcal{L}$).

Příklad 4.5 Uvažujte $\mathbf{L}$ uvedené výše (Łukasiewicz, minimum, součin). Napište příklady formulí, zadejte různá ohodnocení e, určete pravdivostní hodnoty těchto formulí. Najděte formule, které jsou tautologiemi klasické logiky, ale nejsou $\mathbf{L}$-tautologiemi pro Łukasiewicz, minimum, součin.

Při tomto přístupu je možné zcela obdobně, jak jsme provedli ve výrokové logice, zavést pojem důkazu a pojem dokazatelné formule (z dané teorie; teorie je množina formulí; pravidlo *modus ponens* je stejné jako ve výrokové logice, tj. "z φ a $\varphi \Rightarrow \psi$ odvoď ψ"). Věta o korektnosti úplnosti má pak následující tvar: Formule φ je $\mathcal{L}$-tautologií, právě když je dokazatelná z příslušných axiomů (jejich podoba je závislá na $\mathcal{L}$; zde nebudeme rozvádět).

Ohodnocená syntaxe. To je jiný přístup, z hlediska celkového pojetí fuzzy přístupu je přirozenější. Pracujeme zde s jednou pevnou strukturou $\mathbf{L}$ pravdivostních hodnot. *Ohodnocená formule* je dvojice $\langle \varphi, a \rangle$, kde φ je formule podle definice výše (tj. jako u neohodnocené syntaxe) a a je pravdivostní hodnota. Namísto s formulemi se pracuje s ohodnocenými formulemi. Pak teorie je množina ohodnocených formulí a to, že $\langle \varphi, a \rangle$ patří do teorie, říká, že formuli φ považujeme za pravdivou aspoň ve stupni a. Podobně axiomy jsou ohodnocené formule. Poznamenejme, že místo teorie jakožto množiny ohodnocených formulí je možné uvažovat teorie jako fuzzy množiny (neohodnocených) formulí (že $\langle \varphi, a \rangle$ patří do teorie jakožto množiny ohodnocených formulí je stejné jako že formule φ patří do teorie jakožto fuzzy množiny formulí ve stupni a); podobně místo množiny axiomů jako ohodnocených formulí můžeme uvažovat fuzzy množinu axiomů jakožto (neohodnocených) formulí.

Důkaz z teorie je pak posloupnost ohodnocených formulí splňujících podobné podmínky jako důkaz v neohodnoceném pojetí. Speciálně pravidlo *modus ponens* říká "z $\langle \varphi, a\rangle$ a $\langle \varphi \Rightarrow \psi, b\rangle$ odvoď $\langle \psi, a \otimes b\rangle$". Podrobněji, důkaz z množiny A axiomů a z teorie T je posloupnost

$$\langle \varphi_1, a_1\rangle, \ldots, \langle \varphi_n, a_n\rangle$$

splňující pro každé $i = 1, \ldots, n$, že buď (1) $\langle \varphi_i, a_i\rangle$ je axiom (patří do A) nebo (2) $\langle \varphi_i, a_i\rangle$ je předpoklad (patří do T) nebo (3) $\langle \varphi_i, a_i\rangle$ vznikl aplikací odvozovacího pravidla na předchozí prvky důkazu. Končí-li takový důkaz ohodnocenou formulí $\langle \varphi, a\rangle$, říkáme mu důkaz φ ve stupni a. Může se stát (a je to přirozené), že pro φ existuje celá řada důkazů končících $\langle \varphi, a_i\rangle$ (čím větší a_i, tím je ten důkaz lepší; $\langle \varphi, 1\rangle$ znamená, že φ je dokázáno "úplně", tj. že jsme dokázali, že φ je úplně pravdivé). *Stupeň dokazatelnosti* φ z dané teorie je pak definován jako supremum všech a_i, pro která existuje důkaz končící $\langle \varphi, a_i\rangle$. Věta o úplnosti, pak říká, že stupeň dokazatelnosti dané formule je roven stupni tautologičnosti (pravdivosti) dané formule. Přitom stupeň tautologičnosti formule φ je dán jako infimum všech $\|\varphi\|_e$ pro všechna možná $\mathbf{L}$-ohodnocení e.

Úvod do predikátové fuzzy logiky (NEBUDE U ZKOUŠKY POŽADOVÁNO)

syntax, sémantika, fuzzy množiny a fuzzy relace, zejm. binární fuzzy relace na množině, zejm. fuzzy ekvivalence (pro modelování podobnosti)

Úvod do aplikací fuzzy logiky Nejúspěšnějšími aplikacemi fuzzy logiky jsou tzv. fuzzy regulátory a tzv. pravidlové fuzzy systémy. Ty nalezly zejména v Japonsku začátkem 90. let rozsáhlé komerční uplatnění. Dokumentuje to fakt, že slovo "fuzzy" bylo v té době zvoleno v Japonsku slovem roku.

Co je to fuzzy regulátor? Motivace, jednoduchý případ [VIZ PŘEDNÁŠKY]

Poznámka 4.6 Důvody úspěchu: NE - blízkost fuzzy přístupu východnímu myšlení; ANO - povaha japonského trhu (vstřícný novinkám), příznivé okolnosti (nízká cena senzorů), koncepční jednoduchost (umožnila rychlé vzdělání inženýrů)

Mezi nejvýznamnější aplikace pravidlových fuzzy systémů a fuzzy regulátorů patří následující:

- spotřební elektronika (fuzzy pračka (tu je možné zakoupit i v ČR), fuzzy myčka, fuzzy vysavač, fuzzy kamera apod.)
- řízení metra v Japonských městech (fuzzy regulátor zajišťuje plynulé rozjíždění a brždění, lepší než člověk)

- řízení velké průmyslové helikoptéry ovládané hlasem (prof. Michio Sugeno); tuto úlohu se klasickými metodami nepodařilo vyřešit
- řízení velkých průmyslových systémů (např. pece)

Obecněji lze říci, že hlavní aplikace fuzzy logiky tvoří fuzzy relační modelování (pravidlové fuzzy systémy jsou příkladem), tj. modelování pomocí fuzzy relací. Kromě zmíněných pravidlových fuzzy systémů se jedná o

- rozhodování;
- information retrieval a vyhledávání;
- shlukování, rozpoznávání.

Pro omezený rozsah textu se nestihneme věnovat zajímavé oblasti aplikací fuzzy logiky, kterou je fuzzy logické programování. Jde o rozšíření klasického logického programování o principy fuzzy modelování (zejména: databáze faktů může obsahovat fakt s nějakým stupněm pravdivosti, např. fakt `jeUsporny`(`octavia19TDI`) se stupněm 0.9, fakt `jeUsporny`(`octavia14MPI`) se stupněm 0.3). Problematika se rozvíjí.

5 Úvod do modální logiky

Základní motivace. Klasická logika nemá prostředky k formalizaci tvrzení obsahujících modality, např. "je možné, že ...", "je nutné, že ...". Rozšíření klasické logiky, kde toto je možné, se nazývá modální logika. Modální logika našla uplatnění ve formalizaci znalostních systémů a systémů, které pracují s časem (tzv. temporální logika, viz dále).

Základy syntaxe a sémantiky modální logiky Zaměříme se na výrokovou modální logiku. Oproti jazyku klasické výrokové logiky obsahuje jazyk modální logiky navíc unární spojky $\Box$ ($\Box\varphi$ se čte "je možné, že φ") a $\Diamond$ ($\Diamond\varphi$ se čte "je možné, že φ"). Definice formule se příslušným způsobem rozšíří (tj. přidáme pravidlo "je-li φ formule, jsou i $\Box\varphi$ a $\Diamond\varphi$" formule).

Konkrétní význam $\Box\varphi$ může být "je známo, že φ", "věří se, že φ", "vždy v budoucnosti byde platit φ" apod.

Sémantika modální logiky je založena na pojmu možný svět. Možný svět je obecná kategorie (v jednom možném světě může v daný okamžik pršet, ve druhém ne, apod.), která má řadu interpretací. Možné světy mohou být časové okamžiky, mohou reprezentovat názory jednotlivých expertů (co možný svět, to expert) apod.

Definice 5.1 ***Kripkeho struktura*** pro výrokovou modální logiku je trojice $\mathbf{K} = \langle W, e, r\rangle$, kde $W \neq \emptyset$ je množina možných světů, e je zobrazení přiřazující každému $w \in W$ a každému výrokovému symbolu p pravdivostní hodnotu $e(w, p)$ (p je/není pravdivé ve w), $r \subseteq W \times W$ je relace dosažitelnosti ($\langle w, w'\rangle \in r$ znamená, že z w je možné dostat se do w').

Definujeme pravdivostní hodnotu $\|\varphi\|_{\mathbf{K},w}$ formule φ v $\mathbf{K}$ v možném světě w takto: $\|p\|_{\mathbf{K},w} = e(w,p)$ (pro výrokový symbol); $\|\varphi \wedge \psi\|_{\mathbf{K},w} = 1$, právě když $\|\varphi\|_{\mathbf{K},w} = 1$ a $\|\psi\|_{\mathbf{K},w} = 1$ (tedy jako v klasické logice) a podobně pro ostatní výrokové spojky; $\|\Box\varphi\|_{\mathbf{K},w} = 1$, právě když $\|\varphi\|_{\mathbf{K},v} = 1$ pro každý $v \in W$, pro který $\langle w,v\rangle \in r$ (tj. "je nutné, že φ" znamená, že φ je pravdivá v každém možném světě dosažitelném z w); $\|\diamond\varphi\|_{\mathbf{K},w} = 1$, právě když $\|\varphi\|_{\mathbf{K},v} = 1$ pro nějaký $v \in W$, pro který $\langle w,v\rangle \in r$ (tj. "je možné, že φ" znamená, že φ je pravdivá v nějakém možném světě dosažitelném z w).

Ukažte, že $\Box\varphi$ a $\neg\diamond\neg\varphi$ (a duálně) mají stejné pravdivostní hodnoty. Tedy, můžeme začít jen s $\Box$ a definovat $\diamond$ jako odvozenou (nebo naopak).

Příklad 5.2 (1) Pro $r = W \times W$ (každý svět je dosažitelný z každého) se příslušná logika nazývá *logika znalostí* (the logic of knowledge).

(2) Platí-li pro nějaké $W' \subseteq W$, že $r = W \times W'$, nazývá se příslušná logika *logika domnění* (logic of belief).

(3) r je reflexivní, tranzitivní a platí, že pro každé $v, w \in W$ je $\langle v,w\rangle \in r$ nebo $\langle w,v\rangle \in r$. Pak se odpovídající logika nazývá *logika času* (temporální logika, temporal logic, tense logic); $w \in W$ se chápou jako časové okamžiky. Existují ale i jiné systémy logiky času. Poznamenejme, že $\|\Box\varphi\|_{\mathbf{K},w} = 1$ pak znamená, že φ je pravdivá ve všech časových okamžicích počínaje w. Pro r^{-1} pak $\|\Box\varphi\|_{\mathbf{K},w} = 1$ znamená, že φ je pravdivá ve všech časových okamžicích až po w. Co pak znamená $\|\diamond\varphi\|_{\mathbf{K},w} = 1$?

Ukažte, že v případech (1) a (2) nezávisí $\|\varphi\|_{\mathbf{K},w}$ na w (tj. je stejná pro všechna w).

Dále ukažte, že pro (1) mají formule $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$, $\Box\varphi \Rightarrow \Box\Box\varphi$, $\diamond\varphi \Rightarrow \Box\diamond\varphi$, $\Box\varphi \Rightarrow \varphi$ pravdivostní hodnotu 1 pro každé $\mathbf{K}$ a w.

6 Logické programování a Prolog

> A logic program is a set of axioms, or rules, defining relationships between objects. A computation of a logic program is a deduction of consequences of the program. A program defines a set of consequences, which is its meaning. The art of logic programming is constructing concise and elegant programs that have the desired meaning.
>
> *Sterling and Shapiro: The Art of Prolog*

V této části se seznámíme se základy logického programování a programovacího jazyka Prolog. Předpokládá se znalost základních pojmů predikátové logiky. Pokud ji čtenář nemá, může číst dále, ale některým pojmům bude rozumět jen intuitivně.

6.1 Co je logické programování: základní koncepce a historie

Logické programování je jedním z paradigmat programování. Zcela se liší od ostatních paradigmat. Logické programování budeme demonstrovat na příkladě jazyka Prolog.

Prolog vznikl začátkem 70. let ve Francii (Marseille). Významným mezníkem byl japonský projekt počítačů 5. generace, ve kterém byl Prolog vybrán jako základní jazyk logického procesoru centrální jednotky počítače. Přestože japonský projekt nesplnil zcela očekávání, stal se Prolog významným jazykem především pro programování úloh umělé inteligence.

Dnes existuje celá řada implementací Prologu. My se budeme držet základního "jádra", které je společné většině implementací.

Uvidíme, že programovat v Prologu vyžaduje "odlišný způsob myšlení". Proto si budeme před podrobnějším výkladem jazyka Prolog na příkladech ilustrovat jeho základní rysy.

Program a výpočet v logickém programování Základní koncepci logického programování vyjadřuje následující dvojice "rovností":

$$\text{program} \quad = \quad \text{množina axiomů}$$

a

$$\text{výpočet} \quad = \quad \text{konstruktivní důkaz uživatelem zadaného cíle,}$$

nebo volněji: program je souborem tvrzení, kterými programátor (uživatel, expert) popisuje určitou část okolního světa; výpočet nad daným programem, který je iniciován zadáním dotazu, je hledání důkazu dotazu z daného souboru tvrzení.

Poznamenejme, že uvedená koncepce není vůbec nová. Její počátky sahají až k Leibnizovi a jeho projektu tzv. *mathesis universalis*, tj. univerzálního jazyka, ve kterém bude možné zachytit veškerou znalost lidstva. Podle Leibnize by pak mělo být možné vytvořit automatický stroj, který pro dané tvrzení (dotaz) zjistí, zda dotaz logicky vyplývá z daného souboru známých faktů. Teprve výsledky logiky ze začátku 20. století ukázaly meze a možnosti Leibnizovy představy (podrobnější diskuze je nad rámec tohoto textu; čtenáře odkazuji např. na [6]).

Přesto, že je to koncepce přirozená, doznala praktické realizace, jak bylo uvedeno, až v 70. letech 20. století.

Shrňme a doplňme nyní základní rysy logického programování:

- logický program je konečná množina formulí (tvrzení popisující okolní realitu; formule mají speciální tvar)
- výpočet je zahájen zadáním formule-dotazu (tu zadává uživatel)
- cílem výpočtu je najít důkaz potvrzující, že dotaz logicky vyplývá (je dokazatelný) z logického programu (konstruktivnost)

- pokud je takto zjištěno, že dotaz z programu vyplývá, výpočet končí a uživateli je oznámeno `Yes` s hodnotami případných proměnných, které se v dotazu vyskytují
- pokud není zjištěno, že dotaz z programu vyplývá, výpočet končí a uživateli je oznámeno `No`
- může se stát, že výpočet neskočí

Základní rysy (později se o nich přesvědčíme), kterými se logické programování zásadně odlišuje od ostatních programovacích paradigmat:

- programování: programátor specifikuje, *co* se má vypočítat, a ne *jak* se to má vypočítat a *kam* uložit mezivýsledky
- řízení výpočtu: Prolog nemá příkazy pro řízení běhu výpočtu ani pro řízení toku dat, nemá příkazy cyklů, větvení, přiřazovací příkaz
- proměnné: neexistuje rozdělení proměnných na vstupní a výstupní, proměnná může být jednou použita jako vstupní, jindy jako výstupní; proměnná v Prologu označuje během výpočtu objekt, který vyhovuje jistým podmínkám
- nerozlišuje se mezi daty a programem.

Nesmíme však podlehnout lákadlu: je třeba mít na paměti, že programátor není zbaven zodpovědnosti za to, jak bude výpočet probíhat. Výpočet (tj. logické dokazování) je řízen prologovským překladačem a programátor (alespoň chce-li psát efektivní programy) musí pravidla, kterými se výpočet řídí, znát a v souladu s nimi program v Prologu vytvářet.

6.2 Úvod do základních pojmů logického programování

6.2.1 První logický program

Na jednoduchém prologovském programu si ukážeme základní principy a pojmy. Náš první program je následující.

```
muz(petr).
muz(jiri).
muz(vaclav).
muz(pavel).
muz(josef).
zena(jana).
zena(olga).
zena(marie).
zena(tereza).
jeDitetem(petr,jiri).
```

```
jeDitetem(jana,jiri).
jeDitetem(jana,olga).
jeDitetem(vaclav,jiri).
jeDitetem(vaclav,olga).
jeDitetem(jiri,pavel).
jeDitetem(jiri,marie).
jeDitetem(pavel,josef).
jeDitetem(pavel,tereza).
jeSynem(X,Y):-muz(X), jeDitetem(X,Y).
jeDcerou(X,Y):-zena(X), jeDitetem(X,Y).
potomek(X,Y):-jeDitetem(X,Y).
potomek(X,Y):-jeDitetem(Z,Y), potomek(X,Z).
```

Ukážeme: přečteme program, řekneme predikáty, jména predikátů, konstanty, proměnné, kvantifikace.

Dále dotaz: dotaz na fakt bez proměnné, dotaz na fakt s proměnnou, kvantifikace u dotazu, opakované řešení, dotaz na pravidlo bez proměnné a s proměnnou, dále jeDitetem(petr,olga) nevyplyva (ač by se mohlo z kontextu zdát)

6.2.2 Logické programování: fakty, pravidla, dotazy (= Hornovské klazule)

logicky program = konecna mnozina faktu a pravidel;

v Prologu zalezi na jejich usporadani (to je pouzito v algoritmu hledani - viz prace rezolucnho zasobniku)

dalsi typ formuli: dotaz

6.2.3 Co znamená "dokazatelné": rezoluční pravidlo a rezoluční dokazování

strucny uvod

6.2.4 Jak se dokazuje: odstranění nedeterminismu, prologovský překladač a zásobník

K tomu: příklad práce prologovského zásobníku: (1) bez navracení; (2) s jednoduchým navracením

6.2.5 Deklarativní a procedurální sémantika

deklarativni: zalozena na pojmu logickeho vyplyvani

proceduralni: zalozena na SLD-rezoluci, popr. algoritmu prologovskeho zasobniku

6.3 Úvod do jazyka Prolog

6.3.1 Základy syntaxe, termy

6.3.2 Anonymní proměnné

6.3.3 Řízení výpočtu

repeat, fail, cykly, cut, negace

6.3.4 Mimologické predikáty

logicke a aritmeticke operatory
porovnavani termu
rizeni databaze
I/O

6.4 Teoretické základy logického programování

Klauzule, hornovské klauzule Klauzule jsou speciální formule, které hrají v logickém programování zásadní roli. Hornovské klauzule (A. Horn byl významný logik) jsou speciální klauzule, se kterými pracuje Prolog.

Definice 6.1 ***Literál*** je libovolná atomická formule (tzv. pozitivní literál) nebo negace atomické formule (tzv. negativní literál). ***Klauzule*** je libovolná disjunkce literálů. ***Hornovská klauzule*** je klauzule, ve které se vyskytuje maximálně jeden pozitivní literál. Symbolem $\Box$ se označuje prázdná klauzule (tj. klauzule obsahující 0 literálů).

Poznamenejme, že atomické formule se v kontextu logického programování nazývají také atomy.

Poznámka 6.2 (1) $\Box$ je v logickém programování symbolem sporu (viz později). Klauzule jsou totiž speciální formule, a mohou být tedy pravdivé nebo nepravdivé. Chápeme-li $\Box$ jako formuli (podle definice to ale není formule), pak $\Box$ nemůže být nikdy pravdivá (klauzule je disjunkce a disjunkce je pravdivá, právě když je pravdivý aspoň jeden její člen).

(2) Jak uvidíme dále (teď předbíháme, ale usnadní nám to pochopení dalšího výkladu), pracuje Prolog následovně. Je-li dán logický program P zadá-li uživatel dotaz G (popř. obecněji $G_1, \dots, G_n$), překladač Prologu přidá k P negaci dotazu, tj. přidá $\neg\, G$, a snaží se z $P, \neg\, G$ (to je vlastně množina formulí) odvodit (tzv. rezoluční metodou, viz později) spor, tj. odvodit $\Box$. Dá se dokázat, že G sémanticky vyplývá z P ($P \models G$), právě když je z $P, \neg\, G$ odvoditelná $\Box$. Oznámí-li prologovský překladač po zadání dotazu G na program P odpověď `Yes`, znamená to právě, že překladač odvodil z $P, \neg$ klauzuli $\Box$.

Poznámka 6.3 Klauzule $L_1 \vee \cdots \vee L_n$ (L_i jsou literály) se v logickém programování často zapisují jako $\{L_1, \dots, L_n\}$. Pozor, čárky zde neznamenají konjunkce (jak tomu bylo v prologovských pravidlech, ale disjunkce). Tedy $\Box$ se značí $\{\ \}$.

Příklad 6.4 Uvažujme jazyk s unárními relačními symboly *muz* a *zena* binárními relačními symboly *otec* ("být otcem") a *deda* ("být dědem"), symboly konstant (0-ární funkční symboly) *petr*, *pavel*, *jiri*, *vaclav*, *milena*, *jana*.

Formule $muz(petr)$, $zena(jana)$, $zena(pavel)$, $\neg\ zena(pavel)$, $zena(pavel) \vee zena(milena)$, $\neg\ zena(pavel) \vee\neg\ zena(milena)$, $otec(pavel) \vee zena(milena)$, $otec(petr, vaclav)$, $deda(X,Y) \vee\neg\ otec(X,Z) \neg\ otec(Z,Y)$ jsou klauzule. Z nich není $zena(pavel) \vee zena(milena)$ hornovská (ostatní jsou).

Formule $zena(pavel) \wedge zena(milena)$, ani $otec(X,Z) \wedge otec(Z,Y) \Rightarrow deda(X,Y)$ klauzule nejsou (pozor, ta druhá je logicky ekvivalentní klauzuli $deda(X,Y) \vee\neg\ otec(X,Z) \neg\ otec(Z,Y)$, ale sama o sobě klauzulí není).

Co jsou to vlastně hornovské klauzule? Hornovská klauzule je klauzule, která obsahuje nejvýše jeden pozitivní literál. Má tedy jeden z následujících tvarů (dohodněme se, že obecný a existenční uzávěr formule φ budeme značit $\forall\varphi$ a $\exists\varphi$; to je sice poněkud nepřesné, protože obecných i existenčních uzávěrů může existovat více; my ale budeme předpokládat, že všechny volné proměnné jsou seřazeny, např. očíslováním, tj. volné proměnné jsou po řadě $X_1, \dots, X_m$, a obecným uzávěrem rozumíme formuli $(\forall X_1 \cdots \forall X_m)\varphi$ a existenčním uzávěrem rozumíme $(\exists X_1 \cdots \exists X_m)\varphi$):

- (nenulový počet negativních literálů, právě jeden pozitivní literál) $A \vee\neg\ B_1 \vee \cdots \vee\neg\ B_n$ (A, B_i jsou atomické formule); tato klauzule je ekvivalentní formuli $B_1 \wedge \cdots \wedge B_n \Rightarrow A$ (Proč? Uvědomte si, že $B \Rightarrow A$ je ekvivalentní $A \vee\neg\ B$ a že $\neg\ (B_1 \wedge \cdots \wedge B_n)$ je ekvivalentní $\neg\ B_1 \vee \cdots \vee\neg\ B_n$.). Tyto klauzule odpovídají prologovským pravidlům.

- (nulový počet negativních literálů, právě jeden pozitivní literál) A. Tyto klauzule odpovídají prologovským faktům.

- (nenulový počet negativních literálů, žádný pozitivní literál) $\neg\ C_1 \vee \cdots \vee\neg\ C_n$ (C_i jsou atomické formule); tato klauzule, resp. její obecný uzávěr $\forall(\neg\ C_1 \vee \cdots \vee\neg\ C_n)$ je ekvivalentní formuli $\neg\ (\exists C_1 \wedge \cdots \wedge C_n$ (Proč? Uvědomte si, že $\neg\ (\exists x)\varphi$ je ekvivalentní $(\forall x)\ \neg\ \varphi$). Tyto klauzule odpovídají prologovským dotazům.

Definice 6.5 ***Logický program*** (někdy definitní logický program) je konečná množina hornovských klauzulí s jedním pozitivním literálem (tj. klauzulí odpovídajících pravidlům a faktům).

Poznámka 6.6 V logických programech se volné proměnné ve formulích chápou jako univerzálně kvantifikované (to známe z Prologu). Jak je to však s dotazy? Z výše uvedeného víme, že dotaz $C_1, \dots, C_n$ na program P se chápe jako dotaz, zda $P \models C_1, \dots, C_n$ (tj. $P \models C_1 \wedge \ldots \wedge C_n$). Jsou-li ve formulích C_i volné proměnné $X_1, \dots, X_m$, uvažují se jako existenčně kvantifikované, tj. dotazujeme se, zda $P \models (\exists X_1 \cdots \exists X_m) C_1 \wedge \ldots \wedge C_n$ (to dobře známe z Prologu: z pohledu uživatele se ptáme "Existují hodnoty proměnných $X_1, \dots, X_m$ pro které $C_1, \dots, C_n$ vyplývá z programu?"). Víme, že prologovský překladač převádí tento problém na problém, zda z P negace dotazu, tj. z P a $\neg\ (\exists X_1 \cdots \exists X_m) C_1 \wedge \ldots \wedge C_n$, tj. z P (P obsahuje jen fakty a pravidla, a ty jsou ekvivalentní hornovským klauzulím) a $(\forall X_1 \cdots \forall X_m)(\neg\ C_1 \vee \cdots \neg\ C_n)$, je odvoditelný spor, tj. klauzule $\square$. Vidíme tedy, že vše nakonec vede k hledání odvození sporu z univerzálně kvantifikovaných hornovských klauzulí. Hledáme tedy odvození sporu z formulí $\forall H_i$, kde H_i jsou hornovské klauzule (odpovídající faktům, pravidlům a dotazu).

Substituce, unifikace S pojmem substituce jsme se setkali při výkladu syntaxe predikátové logiky. Připomeňme, že $\varphi(x/s)$ a $t(x/s)$ označuje formuli a term, které vzniknou substitucí termu s za proměnnou x podle uvedených pravidel. Výklad o substitucích nyní pro naše potřeby rozšíříme. Substituce můžeme označovat (x/s) a v dalším budeme uvažovat obecně i substituce tvaru $(x_1/s_1, \ldots, x_n/s_n)$ (x_i proměnné, s_i termy), kde proměnné x_i jsou navzájem různé (tj. pro $i \neq j$ je $x_i \neq x_j$) a dále $x_i \neq s_i$. Speciálně, $()$ označuje tzv. prázdnou substituci ($n = 0$, nic se nenahrazuje); pro ni platí $\varphi() = \varphi$ a $t() = t$. Substituce budeme označovat symboly $\theta, \theta_1, \ldots$. Pro substituci $\theta = (x_1/s_1, \ldots, x_n/s_n)$ bude $\varphi\theta$ (popř. $t\theta$) označovat výsledek substituce použité na formuli φ (popř. term t); tento výsledek vznikne současným nahrazením proměnných $x_1, \ldots, x_n$ termy $s_1, \ldots, s_n$ podle stejných pravidel jako v případě $\varphi(x/s)$ (popř. $t(x/s)$).

Budeme potřebovat skládání substitucí. Pro substituce θ a σ budeme symbolem $\sigma \circ \theta$, popř. jen $\sigma\theta$ označovat substituci, která bude mít následující efekt: $\varphi(\sigma\theta)$ bude formule, která vznikne aplikací substituce θ na formuli vzniklou aplikací substituce σ na φ.

Pro substituce $\sigma = (x_1/s_1, \ldots, x_n/s_n)$ a $\theta = (y_1/t_1, \ldots, y_m/t_m)$ definujeme substituci $\sigma\theta$ jako substituci, která vznikne z

$$(x_1/s_1\theta, \ldots, x_n/s_n\theta, y_1/t_1, \ldots, y_m/t_m)$$

vymazáním (1) $x_i/s_i\theta$, pro která $x_i = s_i\theta$ a (2) y_i/t_i, pro která $y_i \in \{x_1, \ldots, x_n\}$.

Lehce se ověří následující tvrzení.

Lemma 6.7 *Pro substituce σ, θ, τ platí (1) $\sigma() = ()\sigma = \sigma$; (2) $E(\sigma\theta) = (E\sigma)\theta$ pro libovolnou formuli, popř. term, E; (3) $(\sigma\theta)\tau = \sigma(\theta\tau)$.*

Tedy: Aplikace $\sigma\theta$ je ekvivalentní postupné aplikaci σ a (potom) θ (bod (2)); substituce je možné skládat, přičemž nezáleží na pořadí skládání (bod (3)).

Definice 6.8 Substituce θ se nazývá ***obecnější*** než substituce σ, jestliže existuje substituce τ, pro kterou $\sigma = \theta\tau$.

Tedy: (přirozený význam) θ je obecnější, protože z ní můžeme dostat τ dodatečným "upřesněním" τ.

Příklad 6.9 Viz přednášky.

Definice 6.10 Substituce θ se nazývá ***unifikační substituce*** (také unifikace) množiny $\{\varphi_1, \ldots, \varphi_n\}$ formulí (popř. termů), pokud $\varphi_i\theta = \varphi_j\theta$ pro libovolné $i, j \in \{1, \ldots, n\}$. Tedy, unifikace je substituce, pop jejíž aplikaci přejdou všechny formule φ_i ve stejnou formuli.

Příklad 6.11 Viz prednasky

Definice 6.12 Unifikace θ množiny $\{\varphi_1, \ldots, \varphi_n\}$ se nazývá ***nejobecnější unfikací*** (zkratka mgu z anglického "most general unifier") této množiny, jestliže je obecnější než každá jiná unifikace množiny $\{\varphi_1, \ldots, \varphi_n\}$.

Poznámka 6.13 mgu neni urcen jednoznacne

Příklad 6.14 Viz přednášky.

Unifikační algoritmus V dalším nazýváme jednoduchým výrazem term nebo atomickou formuli. Cílem je najít k dané množině jednoduchých výrazů nejobecnější unifikaci.

Pro konečnou množinu S jednoduchých výrazů nazvěme její *rozdílovou množinou* následovně definovanou množinu: Najdi první pozici i zleva, na které nemají všechny výrazy z S stejný symbol; podvýrazy všech výrazů z S, které začínají na pozici i tvoří rozdílovou množinu množiny S.

Příklad 6.15 Treba z Lloyda

Popíšeme základní algoritmus pro hledání nejobecnější unifikace dané množiny S jednoduchých výrazů. Pro $S = \{E_1, \ldots, E_n\}$ označme $S\theta = \{E_1\theta, \ldots, E_n\theta\}$.

Algoritmus hledání mgu dané konečné množiny S jednoduchých výrazů:

1. $k := 0$; $\theta_0 := ()$.
2. Je-li $S\theta_k$ jednoprvková, zastav; θ_k je hledaný mgu množiny S. Jinak najdi rozdílovou množinu D_k množiny $S\theta_k$.
3. Existují-li v D_k term t a proměnná x, která se nevyskytuje v t, polož $\theta_{k+1} := \theta_k(x/t)$; $k := k + 1$; jdi na 2. Jinak zastav, S není unifikovatelná.

Příklad 6.16 Treba z Lloyda.

Poznámka 6.17 Tento algoritmus může mít vysokou časovou složitost. Jsou známy algoritmy s lineární časovou složitostí. Ve většině implementací Prologu je však implementována modifikace výše uvedeného algoritmu, která spočívá ve vynechání testu výskytu x v t (viz algoritmus). To však může vést k nesprávným výpočtům.

Obecná rezoluční metoda Popíšeme nyní tzv. obecnou rezoluční metodu navrženou v 60. letech 20. stol. Robinsonem. Základem je tzv. rezoluční odvozovací pravidlo, které ze dvou klauzulí odvodí jinou klauzuli, jejich tzv. rezolventu. Obecný tvar rezolučního pravidla je

> z klauzule $\{L_1, \ldots, L_{i-1}, A, L_{i+1}, \ldots, L_n\}$
> a klauzule $\{L'_1, \ldots, L'_{i-1}, \neg A', L'_{i+1}, \ldots, L'_m\}$
> odvoď klauzuli (tzv. rezolventu výše uvedených klauzulí)
> $\{L_1\theta, \ldots, L_{i-1}\theta, L_{i+1}\theta, \ldots, L_n\theta, L'_1\theta, \ldots, L'_{i-1}\theta, L'_{i+1}\theta, \ldots, L'_m\theta\}$,
> je-li θ mgu množiny $\{A, A'\}$.

Přitom jsou L_i, L'_j literály a A a A' jsou atomické formule.

Poznámka 6.18 Rezoluční pravidlo je svou povahou odvozovací pravidlo ve stejném duchu jako pravidla, se kterými jsme se setkali: z nějakých formulí odvodí jinou formuli. Pravidlo MP lze chápat jako speciální případ rezolučního pravidla: Pravidlo MP říká "z A a $A \Rightarrow B$ odvoď B". Formule $A \Rightarrow B$ je ekvivalentní klauzuli $\neg A \vee B$. Tedy odvození pomocí MP odpovídá odvození $\{B\}$ z $\{\neg A, B\}$ a $\{A\}$. To lze pomocí rezolučního pravidla s unifikací, za kterou vezmeme identickou (prázdnou) substituci.

Příklad 6.19 Vymyslete příklady na rezoluční odvození (1) kdy pravidlo nejde alpikovat, (2) kdy jde jedinym zpusobem, (3) kdy jde vice zpusoby, (4) odvození prázdné klauzule.

Pro danou množinu F klauzulí označme

$$R(F) = \{C \mid C \text{ je rezolventou nějakých klauzulí z } F\}.$$

Dále definujme $R^0(F) = F$ a $R^{n+1}(F) = R^n(F) \cup R(R^n(F))$. Tedy $R^n(F)$ je nmožina klauzulí odvoditelných z F nejvýše n-násobným použitím kroku "utvoř všechny možné rezolventy".

Základní výsledek popisující rezoluční metodu:

Věta 6.20 *Množina F klauzulí je sporná, právě když existuje n tak, že $\Box \in R^n(F)$.*

Poznámka 6.21 Spornost F je třeba chápat jako spornost v dříve uvedeném axiomatickém systému (tj. že z F je dokazatelná libovolná formule). Víme však, že F je sporná, právě když nemá žádný model. Nepotřebujeme se tedy na axiomatický systém odvolávat a vystačíme se sémantickými pojmy.

SLD-rezoluce Víme, že v Prologu je po zadání dotazu G cílem překladače zjistit, zda z programu P dotaz G plyne, tj. zda $P \models G$. Víme, že $P \models G$, právě když $P, \neg G$ je sporná (viz … odkaz). Podle Věty 6.20 je zjištění spornosti

$P, \neg\, G$ ekvivalentní (neboť všechny vyskytující se formule jsou klauzule) zjištění, zda $\Box$ patří do nějakého $R^n(F)$. Výpočet R^n je však obecně náročný a nijak nevyužívá speciální formu klauzulí používanou v Prologu (hornovské klauzule). Cílem následujícího textu je ukázat speciální formu rezoluční metody, tzv. SLD-rezoluci.

V následujcím označuje P konečnou množinu hornovských klauzulí, které odpovídají faktům a pravidlům (tzv. *programové klauzule*, tj. hornovských klauzulí s jedním pozitivním literálem; takové množině se říká logický program, popř. definitní logický program); G označuje hornovskou klauzuli, která odpovídá negaci uživatelem zadaného cíle $? - G_1, \dots, G_n$, tzv. *cílová klauzule*, tj. klauzuli $\neg\, G_1 \vee \cdots \vee \neg\, G_n$.

Definice 6.22 ***Odpověď*** pro $P \cup \{G\}$ je libovolná substituce $\theta = (x_1/t_1, \dots, x_m/t_m)$, kde každá proměnná x_i se vyskytje v G.

Definice 6.23 Odpověď θ pro $P \cup \{G\}$ se nazývá ***korektní*** , jestliže formule $\forall((G_1 \wedge \cdots \wedge G_n)\theta)$ sémanticky plyne z P.

To je základem deklarativní sémantiky logického programu P (je ji možné chápat jako množinu všech G spolu s θ, kde θ je korektní odpověď pro $P \cup \{G\}$; tj. $\mathrm{Decl}(P) = \{\langle ? - G_1, \dots, G_n, \theta\rangle \mid \theta$ je korektní odpověď pro $P \cup \{G\}\})$.

To souvisí s odpověďmi Prologu následovně: Prolog odpoví na dotaz $? - G_1, \dots, G_n$ buď `Yes` se substitucí θ (ta může být prázdná, a pak ji Prolog nevypisuje), nebo `No`. Odpověď `Yes` se substitucí θ je korektní, jestliže θ je korektní odpověď pro $P \cup \{G\}$ v právě definovaném smyslu. Odpověď `No` je korektní, jestliže $\forall((G_1 \wedge \cdots \wedge G_n)\theta)$ sémanticky neplyne z P pro žádnou θ.

Pokoušíme-li se aplikovat rezoluční pravidlo na klauzule, z nichž jedna odpovídá cíli (nemá žádný pozitivní literál) a druhá odpovídá pravidlu nebo faktu (má právě jeden pozitivní literál A), už se nepotýkáme s následujícím nedeterminismem implicitně obsaženým v obecném rezolučním pravidle: z podstaty rezolučního pravidla je jasné, že v cíli hledáme takový G_i, který je možné unifikovat s A (tedy nevybíráme dvojice atomických formulí mající unifikaci jako v obecné rezoluční metodě, ale jen atomické formule, které jsou unifikovatelné s A).

Definice 6.24 Nechť P a G jsou jako výše (logický program a klauzule odpovídající cíli). ***SLD-odvození*** z $P \cup \{G\}$ je posloupnost $H_0 = G, H_1, \dots$ cílových klauzulí, posloupnost $C_1, C_2, \dots$ variant programových klauzulí z P (tj. C_i vzniknou z klauzulí z P přejmenováním proměnných) a posloupnost $\theta_1, \theta_2, \dots$ unifikací takových, že θ_{i+1} je mgu množiny $\{H_i, C_{i+1}\}$ a H_{i+1} je odpovídající rezolventou (tj. vznikne použitím rezolučního pravidla na H_i a C_{i+1} při θ_{i+1}). SLD-odvození se nazývá ***zamítnutí*** (refutace) délky n, jestliže jeho poslední cílová klauzule je $H_n = \Box$.

Poznámka 6.25 To je základem tzv. procedurální sémantiky. SLD-odvození odpovídá výpočtu prologovského překladače. Souvislost s prologovským zásobníkem (téměř zřejmá, stav zásobníku odpovídá SLD-odvození).

Odvození může refutací (úspěšné); nekonečné (neúspěšné 1); konečné nekončící $\Box$, které nelze dále prodloužit (neúspěšné 2).

Definice 6.26 ***Vypočítaná odpověď*** pro $P \cup \{G\}$ je substituce θ, pro kterou existuje refutace z $P \cup \{G\}$ délky n s odpovídající posloupností $\theta_1, \ldots, \theta_n$ takovou, že θ vznikne z $\theta_1 \cdots \theta_n$ (složení substitucí) vynecháním těch x/t, pro které se x nevyskytuje v G.

Věta 6.27 (korektnost SLD-rezoluce) *Každá vypočítaná odpověď pro $P \cup \{G\}$ je korektní odpovědí pro $P \cup \{G\}$.*

Poznámka 6.28 Obsahově odpovídá větě o korektnosti, ...

Věta 6.29 (úplnost SLD-rezoluce; odpověď Yes) *Pro cíl $?-G_1, \ldots, G_n$ a program P platí, že jestliže $\exists(G_1 \curlywedge \cdots \curlywedge G_n)$ sémanticky plyne z P, pak existuje refutace z $P \cup \{G\}$.*

Poznámka 6.30 Odpovídá větě o úplnosti: Jestliže to sémanticky plyne, pak to lze prokázat syntakticky (refutací).

Věta 6.31 (úplnost SLD-rezoluce; odpověď Yes se substitucí) *Pro každou korektní odpověď θ pro $P \cup \{G\}$ existuje vypočítaná odpověď σ pro $P \cup \{G\}$, která je (jako substituce) obecnější než θ (tj. $\theta = \sigma\tau$ pro nějakou substituci τ).*

Poznámka 6.32 Přečtěme to znovu, podrobněji: Jestliže pro cíl $?-G_1, \ldots, G_n$ a program P platí, že $\exists(G_1 \curlywedge \cdots \curlywedge G_n)$ sémanticky plyne z P se substitucí θ, pak to lze prokázat syntakticky refutací, která v sobě má substituci obecnější než θ.

Sjednocující pohled: Uživatelský pohled na PROLOG je založen na deklarativní sémantice (sémantické vyplývání); programátorský pohled je založen na rezolučním zásobníku (tak je hledání odpovědi, tj. hledání refutace se substitucí, tj. hledání důkazu) implementováno; logický pohled je založen na rezoluční metodě, spec. SLD-rezoluci. Výše uvedené výsledky ukazují souvislosti mezi jednotlivými pohledy. Především: Sémantické vyplývání je přirozené z hlediska intuitivního pochopení, co prologovský překladač hledá, jaký je význam odpovědi. Nedává však návod, jak hledání odpovědi implementovat. Toto hledání lze v principu realizovat hledáním důkazu v axiomatickém systému, který byl uveden v základním výkladu o predikátové logice. Takové hledání je však příliš náročné ("příliš nedeterministický" pojem důkazu v axiomatickém systému, tento pojem není primárně vhodný pro hledání důkazu). Proto byla navržena rezoluční metoda, která je ještě vhodnější v případě SLD-rezoluce (viz výše popsaný efekt hornovskosti klauzulí: v cílové klauzuli jsou všechny literály negativní, jediným literálem, který je možné unifikovat v programové klauzuli je tedy jediný pozitivní literál).

Pozor: Úplnost říká, že má-li být odpověď Yes, pak tato odpověď vypočítatelná. To ale neznamená, že ji Prolog vždy najde — překladač prohledává podle algoritmu zásobníku, tj. hledá možné důkazy a může se rozběhnout po nekonečné větvi. Prohledává totiž strom možných řešení do hloubky (při tom využívá očíslování formulí). To však není tím, že bychom zatím špatně hledali a příslušný algoritmus, který refutující výpočet vždy najde, ještě neobjevili. Je to jen jeden z projevů jednoho ze základních výsledků vyčíslitelnosti: predikátová logika je nerozhodnutelná (problém: "zjisti, zda je formule tautologií" je nerozhodnutelný, potažmo pak problém logického vyplývání). Tato situace je analogická situaci, která je více známá, totiž ta, že musíme čelit NP-úplnosti některých problémů. Nejde se jim vyhnout, musíme použít heuristiky. Krása výsledků teoretické informatiky a logiky spočívá v tom, že tyto principiální věci nám sdělují (jinak bychom třeba marně hledali efektivní algoritmy, které neexistují). Z tohoto pohledu je tedy algoritmus prologovského překladače heuristikou čelící nerozhodnutelnosti problému, který nás zajímá ("předpokládám-li znalosti v databázi, plyne z nich můj dotaz?").

Všimněme si dále významu očíslování formulí v prologovském programu, které používá algoritmus prologovského zásobníku. Z pohledu vysvětlené SLD-rezoluce spočívá význam očíslování v tom, že odstraňuje nedeterminismus v kroku, ve kterém je třeba vybrat programovou klauzuli k unifikaci. Zatímco v SLD-rezoluci je připuštěna libovolná možná programová klauzule, algoritmus prologovského zásobníku říká, že se použije ta možná s nejmenším číslem.

7 Dodatek: Potřebné pojmy a výsledky

Obsah dodatku nebude u zkoušky požadován.

Formální jazyky Abeceda je libovolná konečná množina symbolů. Slovo nad danou abecedou je libovolná konečná posloupnost symbolů této abecedy. Uvažujeme také tzv. prázdné slovo (označujeme ho ε), tj. slovo neobsahující žádný symbol. (zřetězení, podslovo, ...) Množinu všech slov nad abecedou Σ se značí Σ^*. Jazyk na danou abecedou je libovolná množina slov této abecedy. (DODELAT)

Algebry (Univerzální) algebra je ... (DODELAT), podalgebra

Strukturální indukce

Věta 7.1 (princip strukturální indukce) *Nechť* $\mathbf{A}$ *je algebra,* $\mathbf{B}$ *její podalgebra generovaná množinou* $C \subseteq A$*. Chceme-li dokázat, že všechny prvky algebry* $\mathbf{B}$ *mají vlastnost* $\mathcal{V}$*, stačí ukázat, že*

(1) všechny prvky z C *mají vlastnost* $\mathcal{V}$*;*

(2) mají-li prvky $b_1, \ldots, b_n \in B$ *vlastnost* $\mathcal{V}$*, pak vlastnost* $\mathcal{V}$ *má také prvek* $f(b_1, \ldots, b_n)$ *pro každou* n*-ární operaci* f *algebry* $\mathbf{A}$*.*

Důkaz. Označme $D_{\mathcal{V}}$ množinu obsahující ty prvky množiny B, které mají vlastnost $\mathcal{V}$, tj. $D_{\mathcal{V}} = \{b \in B \mid b \text{ má } \mathcal{V}\}$. Protože $C \subseteq B$, plyne z (1) $C \subseteq D_{\mathcal{V}}$. Z (2) plyne, že $D_{\mathcal{V}}$ je uzavřená na operace algebry $\mathbf{A}$, a tedy $B \subseteq D_{\mathcal{V}}$ (neboť B je podle předpokladu generovaná množinou C, tj. je nejmenší podmnožinou množiny A, která obsahuje C a je uzavřená na operace algebry $\mathbf{A}$). Důkaz je hotov. $\square$

8 Dodatek: Neformální logika

Reference

[1] Jakákoli učebnice matematické logiky bude vhodným studijním pramenem.

[2] Kolář J., Chytil M., Štěpáknová O.: Logika, algebry a grafy. SNTL, Praha, 1989 (dostupná v knihovnách, učebnice pro VŠ technické).

[3] Lloyd J. W.: Foundations of Logic Programming. Springer, 1984.

[4] Sochor A.: Klasická matematická logika. Karolinum, Praha, 2001 (v prodeji, velmi dobře psaná s řadou doplňujících informací).

[5] Sterling L., Shapiro E.: The Art of Prolog. MIT Press, 1986.

[6] Švejdar V.: Logika, neúplnost a složitost. Academia, Praha, 2002.