INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Logika: systémový rámec rozvoje oboru v ČR a koncepce logických propedeutik pro mezioborová studia (reg. č. CZ.1.07/2.2.00/28.0216, OPVK)

Úvod do logiky (VL): 5. Odvození výrokových spojek z jiných výrokových spojek

doc. PhDr. Jiří Raclavský, Ph.D.

(raclavsky@phil.muni.cz)

5. Odvození výrokových spojek z jiných výrokových spojek

V této kapitole se naučíme, jak si z určité množiny výrokových spojek odvodit všechny zbývající výrokové spojky. Množiny, které takovéto odvození umožňují, se nazývají *funkčně úplné množiny spojek*. Příklady takovýchto množin jsou {¬,→}, {¬,∧}, {¬,∨}, {¬, if-then-else}, anebo dokonce jednoprvkové množiny {↑}, {↓}; funkčně neúplnými jsou např. {∧,∨}, {¬}.

Známá množina pěti výrokových spojek {¬,∧,∨,→,↔} tedy obsahuje redundantní prvky. Ty jsou však užitečné prakticky, poněvadž mají často užívané koreláty v našem jazyce, totiž příslušné gramatické spojky. V logických textech jsou ony ‚nadbytečné' spojky zaváděny pomocí definic jako např. „$p \rightarrow q =_{df} \neg p \vee q$". Taková definice vlastně říká, že „$p \rightarrow q$" je jazyková zkratka za „$\neg p \vee q$". Prakticky však jde o to naznačit, že některé spojky (zde např. ∨) jsou v příslušném dedukčním systému chápány jako základní, kdežto jiné (např. → v našem příkladu) jako odvozené. Množina pravdivostních funkcí se nijak nemění, ta je již dána; odlišení základních a odvozených spojek pouze ukazuje, pomocí jakých výrazů a jakým způsobem o nich budeme mluvit. (Níže v této kapitole nebudeme příliš mezi pravdivostními funkcemi a je označujícími výrokovými spojkami odlišovat.)

Zejména v našem prvním příkladu si ukážeme, jak určitou pravdivostní funkci vyjádřit pomocí formule obsahující pouze zavedené výrokové spojky. Tato metoda je mimochodem zvlášť užitečná, pokud si chceme odvodit nějakou tautologii tvaru ekvivalence, na níž jsme pozapomněli.

5.1 Příklady – odvození výrokových spojek z jiných výrokových spojek

1)

Provedeme odvození nejznámějších výrokových spojek z negace a disjunkce, tj. z množiny {¬,∨}. Nejprve s pomocí ¬ a ∨ odvodíme implikaci. Implikace je binární funkce, proto ji zkusme odvodit z binární funkce disjunkce:

p	∨	q
1	1	1
1	1	0
0	1	1
0	0	0

Zkusíme-li negovat p dostaneme definici námi hledané implikace →:

$\neg$	p	$\vee$	q
0	1	*1*	1
0	1	*0*	0
1	0	*1*	1
1	0	*1*	0

Obdobným způsobem odvodíme konjunkci. Tu lze odvodit buď z právě zavedené implikace, anebo z disjunkce, jež je základnější. V námi již známé formuli $\neg p \vee q$ zkusíme negovat též q, což dává:

$\neg$	p	$\vee$	$\neg$	q
0	1	*0*	0	1
0	1	*1*	1	0
1	0	*1*	0	1
1	0	*1*	1	0

Vidíme, že výsledný sloupec pravdivostních hodnot je ‚zrcadlovým obrazem' hledaného průběhu pravdivostních podmínek. Proto negujeme celou formuli – dostaneme tak konjunkci $\wedge$:

$\neg($	$\neg$	p	$\vee$	$\neg$	$q)$
1	0	1	0	0	1
0	0	1	1	1	0
0	1	0	1	0	1
0	1	0	1	1	0

Ekvivalenci $\leftrightarrow$ nyní pro jednoduchost definujeme jako implikaci oběma směry:

$(p$	$\rightarrow$	$q)$	$\wedge$	$(q$	$\rightarrow$	$p)$
1	1	1	*1*	1	1	1
1	0	0	*0*	0	1	1
0	1	1	*0*	1	0	0
0	1	0	*1*	0	1	0

2)

Nyní odvodíme nejznámější pravdivostní funkce s pomocí Schefferovy funkce:

p	$\uparrow$	q
1	*0*	1
1	*1*	0
0	*1*	1
0	*1*	0

Základní krok spočívá v tom, že si pomocí $\uparrow$ definujeme negaci, což vypadá obtížně vzhledem k tomu, že $\uparrow$ je binární, kdežto $\neg$ unární. Trik bude v tom, že $\uparrow$ nebude operovat na všech možných dvojicích pravdivostních hodnot, ale jen na některých:

p	$\uparrow$	p
1	*0*	1
0	*1*	0

Čili $\neg p$ je definovatelná pomocí $p \uparrow p$. Pro srovnání uvádíme odvození pravdivostní funkce unární verum:

$(p$	$\uparrow$	$p)$	$\uparrow$	p
1	0	1	*1*	1
0	1	0	*1*	0

S pomocí negace snadno definujeme konjunkci $\wedge$:

$\neg$	$(p$	$\uparrow$	$q)$
1	1	0	1
0	1	1	0
0	0	1	1
0	0	1	0

Už v předchozím příkladu jsme viděli, jak třeba z $\neg$ a $\wedge$ nadefinovat postupně $\vee$, $\rightarrow$, $\leftrightarrow$. Nejznámější výrokové spojky ale lze nadefinovat výlučně s pomocí Schefferovy funkce $\uparrow$. Definici $\neg$ jsme již viděli; nyní definujme třeba $\wedge$, ovšem bez využití $\neg$:

$(p$	$\uparrow$	$q)$	$\uparrow$	$(p$	$\uparrow$	$q)$
1	0	1	*1*	1	0	1
1	1	0	*0*	1	1	0
0	1	1	*0*	0	1	1
0	1	0	*0*	0	1	0

Implikaci $\rightarrow$ definujeme takto:

$(p$	$\uparrow$	$q)$	$\uparrow$	p
1	0	1	*1*	1
1	1	0	*0*	1
0	1	1	*1*	0
0	1	0	*1*	0

Nyní definujeme pro ukázku vylučovací disjunkci $\vee\!\vee$:

((p	$\uparrow$	q)	$\uparrow$	p)	$\uparrow$	((p	$\uparrow$	q)	$\uparrow$	q)
1	0	1	1	1	*0*	1	0	1	1	1
1	1	0	0	1	*1*	1	1	0	1	0
0	1	1	1	0	*1*	0	1	1	0	0
0	1	0	1	0	*0*	0	1	0	1	0

Definice disjunkce $\vee$ je jednodušší:

(p	$\uparrow$	p)	$\uparrow$	(q	$\uparrow$	q)
1	0	1	*1*	1	0	1
1	0	1	*1*	0	1	0
0	1	0	*1*	1	0	0
0	1	0	*0*	0	1	1

Konečně ekvivalenci $\leftrightarrow$ definujeme pomocí $\uparrow$ spojující formuli pro $\vee$ s formulí $p\uparrow q$:

((p	$\uparrow$	p)	$\uparrow$	(q	$\uparrow$	q))	$\uparrow$	(p	$\uparrow$	q)
1	0	1	1	1	0	1	*1*	1	0	1
1	0	1	1	0	1	0	*0*	1	1	0
0	1	0	1	1	0	0	*0*	0	1	1
0	1	0	0	0	1	1	*1*	0	1	0

3)

O něco komplikovanější je definování tří- a vícečetných pravdivostních funkcí. Tento podnik je založen na tom, že každé formuli koresponduje nějaká pravdivostní funkce. Z toho plyne, že formuli obsahující například tři (odlišné) proměnné odpovídá nějaká ternární funkce. (V některých případech je například třetí proměnná eliminovatelná, čemuž je tak tehdy, když hodnotu nějaké ternární funkce lze definovat zcela bez pomoci třetího parametru.) Příkladem je už výše uváděná definice funkce if-then-else pomocí formule $(p\rightarrow q)\wedge(\neg p\rightarrow r)$.