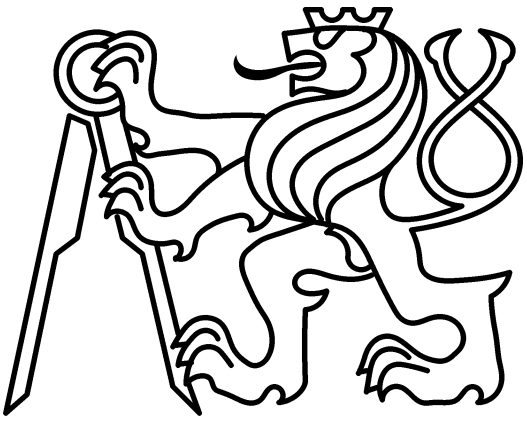
**České Vysoké Učení Technické v Praze**
**Fakulta elektrotechnická**

# Logika programů

Jiří Velebil
katedra matematiky
Praha, 1999

velebil@math.feld.cvut.cz
http://math.feld.cvut.cz/velebil



Vážený čtenáři,

text, který máte před sebou, slouží jako studijní materiál k výběrové přednášce *Logika programů* na fakultě elektrotechnické ČVUT v Praze. Děkuji všem studentům letního semestru roku 1999. Jejich připomínky měly na obsah a uspořádání textu významný vliv. Za případné chyby a nejasnosti, které v textu jsou, nesu plnou zodpovědnost já sám.

Budu proto velmi vděčen za jakékoli další připomínky týkající se struktury textu, jeho čitelnosti a správnosti.

Jiří Velebil

# Obsah


**1 O čem je tento text** **6**
- 1.1 Výroková logika a modální výroková logika . . . 6
- 1.2 Co je logika programů a procesů . . . 8
- 1.3 Obsah jednotlivých kapitol . . . 9

**2 Rychlé repetitorium** **13**
- 2.1 Binární relace a grafy . . . 14
- 2.2 Cvičení . . . 17

**3 Procesy a akční struktury** **19**
- 3.1 Akční struktury . . . 19
- 3.2 Zadávání akčních struktur . . . 20
- 3.3 Morfismy akčních struktur . . . 22
- 3.4 Cvičení . . . 25

**4 Modální logika** **27**
- 4.1 Hennessyho-Milnerova modální logika . . . 27
- 4.2 Sémantika modálních formulí . . . 29
- 4.3 Morfismy modelů modální logiky . . . 30
- 4.4 Cvičení . . . 32

**5 Systémy modální logiky** **35**
- 5.1 Systém **K** — minimální modální logika . . . 35
- 5.2 Věta o korespondenci . . . 38
- 5.3 Modální definovatelnost . . . 39
- 5.4 Cvičení . . . 42

**6 Parciální korektnost programů** **44**
- 6.1 Jazyk `while` programů . . . 44
- 6.2 Zamýšlený modální model . . . 45
- 6.3 Formule parciální korektnosti . . . 46
- 6.4 Deduktivní systém formulí parciální korektnosti . . . 47
- 6.5 Cvičení . . . 52

**7 Modální logika jako algebra** **54**
- 7.1 Částečně uspořádané množiny . . . 54
- 7.2 Výroková logika jako algebra . . . 55
- 7.3 Modální algebry . . . 57
- 7.4 Korespondence Booleových algeber a modelů výrokové logiky . . . 58
- 7.5 Korespondence modálních algeber a modálních modelů . . . 61
- 7.6 Věty o úplnosti . . . 62
- 7.7 Cvičení . . . 64

**8 Modální $\mu$-kalkulus** **68**
8.1 Neadekvátnost modální logiky . . . 68
8.2 Ještě jednou nejslabší liberální prepodmínka . . . 69
8.3 Pevné body v úplných svazech . . . 72
8.4 Dualita pevných bodů . . . 76
8.5 Motivace syntaxe a sémantiky $\mu$-modálních formulí . . . 77
8.6 Vlastnosti Liveness a Safety . . . 84
8.7 Konečné aproximace modálních $\mu$-formulí . . . 86
8.8 Cvičení . . . 90

**9 Jazyk CCS** **92**
9.1 Syntaxe CCS . . . 92
9.2 Silná bisimilarita procesů . . . 98
9.3 Přejmenovávání komunikačních portů . . . 102
9.4 Uzavřené procesy . . . 103
9.5 Cvičení . . . 104

**Literatura** **107**

# Kapitola 1

# O čem je tento text

Text. který začínáte číst, je věnován aplikaci klasické matematické disciplíny — *modální logiky* — na problematiku chování programů a obecněji na problematiku chování procesů.

Klasická modální logika je součástí matematiky, přesněji součástí té partie matematiky, která studuje nejrůznější *formální* logické jazyky. V dalších odstavcích vysvětlíme základní pojmy a možnosti klasické modální výrokové logiky a naznačíme, jakým způsobem lze standardní definice zobecnit na studium procesů.

## 1.1 Výroková logika a modální výroková logika

Připomeňme, že potřeba *výrokové logiky* tak, jak je vyložena například ve skriptu [DePo97], vznikla z nejasností, které jsou způsobeny používáním přirozeného jazyka pro popis nejrůznějších (i velmi jednoduchých) situací, jakou je například výrok

*Mám hlad a chce se mi spát.*

Tyto problémy lze odstranit tak, že přesně vymezíme, co je výrok a jakým způsobem lze nové výroky vytvářet.

1. Některé věty deklarativně za výroky prohlásíme. Těmto větám budeme říkat *atomické výroky*, protože je hodláme považovat za dále nedělitelné.
2. Další výroky budeme z již zkonstruovaných výroků vytvářet předem dohodnutým způsobem. Konstrukce, které budeme k vytváření nových výroků používat, nazveme *logické spojky*.
3. Žádným jiným způsobem než výše popsanými *nelze* výrok zkonstruovat.

Při určování pravdivosti výroků budeme postupovat následovně:

1. Zjistíme, zda daný výrok je atomický nebo ne.
2. Pro atomické výroky musíme jejich pravdivost či nepravdivost *znát*.
3. Není-li výrok atomický, pak vznikl spojením jednodušších výroků. Známe-li pravdivost jednodušších výroků a *rozumíme-li* jednotlivým spojkám, zjistíme pravdivost složitějšího výroku.

Snadno zjišťujeme, že přirozený jazyk lze v úvahách z předchozích odstavců nahradit *symboly* a zavést tak výrokovou logiku čistě formálně. Namísto *výrok* budeme říkat *formule*.

1. Zvolíme množinu $At$. Jejím prvkům budeme říkat *atomické formule*.
2. Zvolíme množinu $S$ disjunktní s $At$. Prvkům množiny $S$ budeme říkat *logické spojky*. Každé logické spojce přiřadíme kladné přirozené číslo — její *aritu*. Arita spojky udává, kolik formulí daná spojka spojí v novou formuli. Běžným přístupem je volba spojek

   (a) *Negace*: $\neg$ arity 1.
   (b) *Konjunkce*: $\wedge$ arity 2.
   (c) *Disjunkce*: $\vee$ arity 2.
   (d) *Implikace*: $\Rightarrow$ arity 2.
   (d) *Ekvivalence*: $\Leftrightarrow$ arity 2.

*Obecná formule* je pak buď atomická formule, nebo řetězec $(\neg\varphi)$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \Rightarrow \psi)$, $(\varphi \Leftrightarrow \psi)$, kde $\varphi$ i $\psi$ jsou formule.

3. Žádným jiným způsobem než konečným použitím výše popsaných pravidel *nelze* formuli zkonstruovat.

Sémantika takto vytvořených formulí je dána jejich *ohodnocením*. Ohodnocení formule nabývá hodnoty buď 1 (pravda) nebo 0 (nepravda). Přitom chceme, aby byla ohodnocena *každá* vytvořená formule. Ohodnocení je tedy funkce $u$ z množiny všech formulí do množiny $\{0, 1\}$ taková, že jsou splněny následující požadavky:

1. $u(\neg\varphi) = 1$ právě tehdy, když $u(\varphi) = 0$.
2. $u(\varphi \wedge \psi) = 1$ právě tehdy, když $u(\varphi) = u(\psi) = 1$.
3. $u(\varphi \vee \psi) = 0$ právě tehdy, když $u(\varphi) = u(\psi) = 0$.
4. $u(\varphi \Rightarrow \psi) = 1$ právě tehdy, když $u(\varphi) = 1$ a $u(\psi) = 0$.
5. $u(\varphi \Leftrightarrow \psi) = 1$ právě tehdy, když $u(\varphi) = u(\psi)$.

Je zřejmé, že pokud pro každou atomickou formuli $a$ zadáme hodnotu $u(a)$, pak výše uvedená pravidla nám umožní zjistit hodnotu $u(\varphi)$ pro libovolnou formuli $\varphi$ výrokové logiky.

Takto definovaná syntaxe a sémantika výrokové logiky umožňuje formální popis základních logických konstrukcí, jako je například otázka správnosti úsudku.

Nevýhody výrokové logiky spočívají v její malé vyjadřovací bohatosti. Jednou nevýhodou je, že jazyk výrokové logiky nepřipouští práci s proměnnými. Některá důležitá tvrzení, jako například větu

*Ne každý, kdo hraje na housle, je Sherlock Holmes.*

tak nelze ve výrokové logice adekvátně popsat. Zobecnění jazyka výrokové logiky, ve kterém lze výše uvedenou větu popsat formálně, se nazývá *predikátová logika*.

Jinou nevýhodou jazyka výrokové logiky je fakt, že neumožňuje studovat *modality* výroků. V jazyce výrokové logiky tak nelze adekvátně popsat například větu

*Je možné, že se mi chce spát.*

Zobecněním výrokové logiky, které umožňuje výroky modalisovat, je *modální výroková logika*.

V modální výrokové logice připouštíme, aby každá formule mohla mít dvě modality: *je nutné, že* a *je možné, že*. Tyto dvě modality jsou vlastně novými logickými spojkami arity 1. Budeme je značit $\Box$ a $\Diamond$.

Definice syntaxe formulí modální výrokové logiky je následující:

1. Zvolíme množinu $At$. Jejím prvkům budeme říkat *atomické formule*.
2. Zvolíme množinu $S$ disjunktní s $At$. Prvkům množiny $S$ budeme říkat *logické spojky*. Každé logické spojce přiřadíme kladné přirozené číslo — její *aritu*. Arita spojky udává, kolik formulí daná spojka spojí v novou formuli.

   (a) *Negace*: $\neg$ arity 1.
   (b) *Konjunkce*: $\wedge$ arity 2.
   (c) *Disjunkce*: $\vee$ arity 2.
   (d) *Implikace*: $\Rightarrow$ arity 2.
   (e) *Ekvivalence*: $\Leftrightarrow$ arity 2.
   (f) *Box*: $\Box$ arity 1.
   (g) *Diamant*: $\Diamond$ arity 1.

   *Obecná formule modální výrokové logiky* je pak buď atomická formule, nebo řetězec $(\neg\varphi)$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \Rightarrow \psi)$, $(\varphi \Leftrightarrow \psi)$, $(\Box\varphi)$, $(\Diamond\varphi)$, kde $\varphi$ i $\psi$ jsou formule.
3. Žádným jiným způsobem než konečným použitím výše popsaných pravidel *nelze* formuli modální výrokové logiky zkonstruovat.

Definice syntaxe formulí modální výrokové logiky se tedy příliš neliší od definice syntaxe výrokové logiky. Definice sémantiky se však budou lišit velmi dramaticky.

Formule modální výrokové logiky opět chceme interpretovat jako buď pravdivé nebo nepravdivé. Naše definice však umožňuje vytvořit formule i pomocí modalit a my chceme formuli $(\Box\varphi)$ číst jako *je nutné, že* $\varphi$ a formuli $(\Diamond\varphi)$ jako *je možné, že* $\varphi$. Co ale znamená říci, že formule *je možné, že* $\varphi$ je pravdivá?

Řešení nabízí takzvaná *sémantika možných světů*. Představme si, že je dána neprázdná množina $W$, jejím prvkům budeme říkat *světy*. Pravdivost formule modální výrokové logiky budeme definovat vždy v jednom konkrétním světě $w \in W$. Dále budeme předpokládat, že pro tento svět je zadána (třeba i prázdná) množina světů $R(w)$. Světům z množiny $R(w)$ říkáme *světy relevantní pro* $w$. Intuice je zde taková, že některé světy z množiny $W$ jsou natolik odlišné od světa $w$, že na pravdivost formule ve světě $w$ nemohou mít žádný vliv — to jsou světy z množiny $W \setminus R(w)$. Naopak světy z množiny $R(w)$ v nějakém smyslu pro pravdivost formule ve světě $w$ podstatné jsou.

Pravdivost formule $\varphi$ modální výrokové logiky je pak ve světě $w$ definována následovně:

1. Formule $(\neg\varphi)$ je pravdivá ve světě $w$ právě tehdy, když formule $\varphi$ ve světě $w$ pravdivá není.
2. Formule $(\varphi \wedge \psi)$ je pravdivá ve světě $w$ právě tehdy, když obě formule $\varphi$ a $\psi$ jsou ve světě $w$ pravdivé.
3. Formule $(\varphi \vee \psi)$ je pravdivá ve světě $w$ právě tehdy, když alespoň jedna z formulí $\varphi$ a $\psi$ je ve světě $w$ pravdivá.
4. Formule $(\varphi \Rightarrow \psi)$ je pravdivá ve světě $w$ právě tehdy, když z pravdivosti formule $\varphi$ ve světě $w$ plyne pravdivost formule $\psi$ ve světě $w$.
5. Formule $(\varphi \Leftrightarrow \psi)$ je pravdivá ve světě $w$ právě tehdy, když formule $\varphi$ je pravdivá ve světě $w$ přesně tehdy, když je ve světě $w$ pravdivá formule $\psi$.
6. Formule $(\Box\varphi)$ je pravdivá ve světě $w$ právě tehdy, když formule $\varphi$ je pravdivá ve všech světech $w'$ z množiny $R(w)$.
7. Formule $(\Diamond\varphi)$ je pravdivá ve světě $w$ právě tehdy, když formule $\varphi$ je pravdivá v alespoň jednom světě $w'$ z množiny $R(w)$.

Je zřejmé, že pokud pro každou atomickou formuli $a$ zadáme její pravdivostní hodnotu ve světě $w$, pak výše uvedená pravidla nám umožní zjistit pravdivost či nepravdivost libovolné formuli modální výrokové logiky ve světě $w$.

Takto definovanou syntaxi a sémantiku modální výrokové logiky v tomto textu zobecníme na takzvanou *Hennessyho-Milnerovu modální výrokovou logiku*. V příštím odstavci naznačíme, o co jde.

## 1.2 Co je logika programů a procesů

V tomto odstavci přiblížíme definici Hennessyho-Milnerovy modální výrokové logiky. Tato logika bude vhodným nástrojem pro popis chování programů a procesů. Nejprve však přeformulujeme sémantiku možných světů modální výrokové logiky tak, že přechod k Hennessyho-Milnerově modální výrokové logice bude velmi přirozený.

V množině možných světů $W$ jsme pro každé $w$ zadali množinu relevantních světů $R(w)$. To lze říci jinak: na množině $W$ je zadána binární relace $R$ tak, že světy $w$ a $w'$ jsou v relaci $R$ přesně tehdy, když $w' \in R(w)$. Binární relaci na množině lze však graficky znázornit — jde o orientovaný graf. Množinou vrcholů je $W$ a vrcholy $w$ a $w'$ jsou spojeny hranou vedoucí z $w$ do $w'$ přesně tehdy, když platí $w\ R\ w'$.

Orientovaný graf si lze představit dynamicky jako automat, který má vrcholy grafu jako stavy a který z jednoho stavu do druhého přechází po orientované hraně grafu. Množina $R(w)$ je při tomto pohledu množina všech stavů automatu, kterých lze ze stavu $w$ dosáhnout v jednom výpočetním kroku.

Podívejme se, jak při tomto pohledu vypadá sémantika možných světů modální výrokové logiky. Namísto svět budeme říkat stav:

1. Formule $(\neg\varphi)$ je pravdivá ve stavu $w$ právě tehdy, když formule $\varphi$ ve stavu $w$ pravdivá není.
2. Formule $(\varphi \wedge \psi)$ je pravdivá ve stavu $w$ právě tehdy, když obě formule $\varphi$ a $\psi$ jsou ve stavu $w$ pravdivé.
3. Formule $(\varphi \vee \psi)$ je pravdivá ve stavu $w$ právě tehdy, když alespoň jedna z formulí $\varphi$ a $\psi$ je ve stavu $w$ pravdivá.
4. Formule $(\varphi \Rightarrow \psi)$ je pravdivá ve stavu $w$ právě tehdy, když z pravdivosti formule $\varphi$ ve stavu $w$ plyne pravdivost formule $\psi$ ve stavu $w$.

5. Formule $(\varphi \Leftrightarrow \psi)$ je pravdivá ve stavu $w$ právě tehdy, když formule $\varphi$ je pravdivá ve stavu $w$ přesně tehdy, když je ve stavu $w$ pravdivá formule $\psi$.

6. Formule $(\Box\varphi)$ je pravdivá ve stavu $w$ právě tehdy, když formule $\varphi$ je pravdivá ve všech stavech $w'$ dosažitelných přechodem ze stavu $w$ v jednom výpočetním kroku.

7. Formule $(\Diamond\varphi)$ je pravdivá ve stavu $w$ právě tehdy, když formule $\varphi$ je pravdivá v alespoň jednom stavu $w'$ dosažitelného ze stavu $w$ v jednom výpočetním kroku.

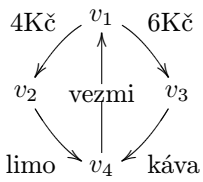
Je zřejmé, že pokud pro každou atomickou formuli $a$ zadáme její pravdivostní hodnotu ve stavu $w$, pak výše uvedená pravidla nám umožní zjistit pravdivost či nepravdivost libovolné formuli modální výrokové logiky ve stavu $w$.

Zjišťujeme, že sémantika možných světů pro modální výrokovou logiku je vhodným nástrojem pro popis chování jednoduchého automatu. Pravdivost formule $(\Box\varphi)$ ve stavu $w$ tak například znamená, že $\varphi$ platí ve všech stavech, do kterých se automat může dostat v jednom výpočetním kroku ze stavu $w$.

Hennessyho-Milnerova modální výroková logika vzniká dotažením výše uvedeného příkladu do realističtější podoby. V automatu, který jsme popsali, nelze jednotlivé druhy výpočetních kroků nijak rozlišovat. Takové rozlišení je ale při studiu automatů velmi důležité: příkladem může být situace, kdy automat čte data, zapisuje data a komunikuje s jiným automatem, pak máme k dispozici tři různé druhy výpočetních kroků — později budeme říkat tři různé *akce*.

Rozlišení jednotlivých druhů výpočetních kroků lze dosáhnout, pokud budeme na jedné množině mít k dispozici několik orientovaných grafů současně. Uvedeme příklad.

Následující obrázek je obrázkem jednoduchého automatu na kávu a limonádu ([Mi89]):




(1.1)

Tento automat má čtyři stavy $v_1$, $v_2$, $v_3$ a $v_4$. Může provádět akce pěti druhů: 4Kč (vhození čtyř korun), 6Kč (vhození šesti korun), limo (volba limonády), káva (volba kávy) a vezmi (odebrání nápoje).

Hennessyho-Milnerova modální výroková logika nám umožní o jednotlivých stavech pronášet výroky typu:

*Je-li automat ve stavu $v_1$, pak lze vhodit čtyři koruny, ale nelze odebrat nápoj.*

Prostředky Hennessyho-Milnerovy modální výrokové logiky lze velmi dobře popsat *konečné* chování procesů. Bohužel v ní nelze adekvátně vyjádřit některá tvrzení o *nekonečném* chování, například větu:

*Automat jednou přejde do stavu, kdy lze odebrat nápoj.*

Rozšíření Hennessyho-Milnerovy modální výrokové logiky, které je vhodné pro popis nekonečného chování, se nazývá *modální $\mu$-kalkulus*. V tomto textu zavedeme modální $\mu$-kalkulus v kapitole 8. Bohužel definice syntaxe a sémantiky formulí modálního $\mu$-kalkulu jsou technicky náročné, proto v této úvodní kapitole o těchto tématech nemůžeme říci více.

## 1.3 Obsah jednotlivých kapitol

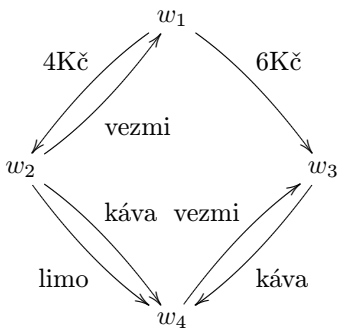
Text je řazen do kapitol podle jednotlivých témat a na konci každé kapitoly je řada cvičení určených k samostatnému zpracování. Doporučujeme, aby čtenář tato cvičení řešil — jejich výsledky jsou často v dalším textu použity.

**Kapitola 2** Tato kapitola slouží k připomenutí známých pojmů jako jsou *binární relace* a *graf*. Důraz je kladen na kategoriální vlastnosti grafů, tj. vysvětlíme například pojmy *součin grafů*, *podgraf* a *koprodukt grafů*.

**Kapitola 3** Zde jsou zavedeny dva základní pojmy tohoto textu — *akční struktura* a *morfismus akčních struktur*. Akční struktura je „více orientovaných grafů na téže množině“. Protože si orientovaný graf na množině

představujeme dynamicky jako automat, ve kterém nelze rozlišit druhy jednotlivých výpočetních kroků, modeluje akční struktura situaci, kdy jednotlivé druhy výpočetních kroků rozlišit umíme. Namísto druh výpočetního kroku budeme říkat *jméno akce*. Množinu jmen akcí $\Delta$ lze chápat jako *specifikaci* typů akcí, které mají automaty provádět.

Akční strukturu typu $\Delta$ potom lze chápat jako *implementaci* specifikovanou množinou $\Delta$. Pokud je například $\Delta = \{4Kč, 6Kč, káva, limo, vezmi\}$, pak je automat z obrázku (1.1) implementací množiny $\Delta$. Naše definice akční struktury ovšem připouští, aby i automat




(1.2)

byl akční strukturou typu $\Delta$.

Akční struktury hodláme navzájem porovnávat pomocí morfismů mezi nimi. Proto v této kapitole zavedeme i pojem *zig-zag morfismu* akčních struktur. Tento pojem přirozeně vzniká ze studia *zig-zag relací*. Zig-zag relace hrají (v nejrůznějších podobách) při studiu logiky programů a procesů velmi důležitou úlohu. Lze například vysvětlit, proč jsou automaty (1.1) a (1.2) různé — neexistuje totiž mezi nimi žádná netriviální zig-zag relace. Část kapitoly 3 je tedy věnována motivačním úvahám, které definici zig-zag relace vysvětlují.

Malá část kapitoly 3 je věnována notaci *jazyka CCS*. Tato notace umožní elegantní zadávání jednotlivých akčních struktur. Notaci jazyka CCS nejprve zavedeme neformálně, formálnímu popisu jazyka CCS je věnována celá kapitola 9.

**Kapitola 4** V této kapitole je zavedena syntaxe a sémantika Hennessyho-Milnerovy modální výrokové logiky. Syntaxe formulí této logiky je zobecněním syntaxe formulí modální výrokové logiky následujícím způsobem: jazykem Hennessyho-Milnerovy logiky chceme mluvit o akčních strukturách typu $\Delta$, proto namísto dvou modalit $\Box$ a $\Diamond$ zavedeme modality $[R]$ a $\langle R\rangle$ pro každé $R \in \Delta$.

Sémantika formulí Hennessyho-Milnerovy modální logiky je pak přímým zobecněním sémantiky možných světů pro modální výrokovou logiku. Znamená to, že formule Hennessyho-Milnerovy logiky budeme interpretovat v akčních strukturách. Platnost výroku

> *Je-li automat ve stavu $v_1$, pak lze vhodit čtyři koruny, ale nelze odebrat nápoj.*

o automatu (1.1) tak budeme moci nahradit rčením

> *Formule* $\langle$4Kč$\rangle$`tt` $\wedge$ [vezmi]`ff` *je pravdivá ve vrcholu* $v_1$.

**Kapitola 5** Některé formule Hennessyho-Milnerovy modální logiky nemají o akčních strukturách velkou vypovídací schopnost. Například formule [káva]`tt` $\vee \neg$[káva]`tt` je pravdivá nejenom v automatu (1.1), tato formule je pravdivá vůbec v každé akční struktuře typu $\Delta = \{4Kč, 6Kč, káva, limo, vezmi\}$. Formulím, které mají tuto vlastnost, budeme říkat *modální tautologie*. V kapitole 5 pak podáme *syntaktickou charakterisaci* modálních tautologií — takzvaný *deduktivní systém* **K** *minimální modální logiky*. Jedná se o analogii systému přirozené dedukce známého z výrokové logiky.

Zajímavým výsledkem této kapitoly je *věta o korespondenci*. Toto tvrzení nám umožní kvalifikovat některé formule Hennessyho-Milnerovy modální logiky jako „nezajímavé“, pokud je budeme interpretovat v akčních strukturách určitého typu. Tak ukážeme například, že formule $[R]\varphi \Rightarrow [R][R]\varphi$ (kde $\varphi$ je libovolná formule) je pravdivá přesně ve všech akčních strukturách, kde jméno akce $R$ je interpretováno jako transitivní binární relace.

Přirozenou otázkou, která vyplývá z věty o korespondenci, je otázka *modální definovatelnosti*:

> Dovedeme libovolnou třídu akčních struktur popsat modální formulí?

Odpověď na tuto otázku je: obecně nikoli. Této problematice je věnována poslední část kapitoly.

**Kapitola 6** V této kapitole je zavedena syntaxe jednoduchého imperativního programovacího jazyka. Tento jazyk umožňuje tvorbu základní rekursivní konstrukce — cyklu `while`. Výsledný jazyk `while` programů lze interpretovat v akční struktuře určitého typu a dát mu tak operační sémantiku. Vzniká jeden konkrétní modální model $\mathscr{M}$, který si lze představit jako *zamýšlenou implementaci* jazyka `while` programů.

V modálním modelu $\mathscr{M}$ pak lze formulovat otázku *parciální korektnosti programu*. Jde o pohled na program $p$ jako na transformační mechanismus, který transformuje formuli $\varphi$ (to, co je pravda před vykonáním programu $p$) na formuli $\psi$ (to, co je pravda po úspěšném vykonání programu $p$). Jazyk Hennessyho-Milnerovy modální logiky nám umožňuje tuto situaci snadno popsat:

> *Zajímá nás pravdivost formule* $\varphi \Rightarrow [p]\psi$.

V tomto kontextu se formuli $\varphi$ říká *prepodmínka programu* $p$ a formuli $\psi$ říkáme *postpodmínka programu* $p$. Formuli $\varphi \Rightarrow [p]\psi$ pak říkáme *formule parciální korektnosti programu* $p$.

V kapitole 6 zavedeme deduktivní systém pro formule parciální korektnosti programů a ukážeme korektnost a úplnost tohoto deduktivního systému.

Dalším důležitým pojmem pro formule parciální korektnosti je pojem *nejslabší liberální prepodmínky programu* $p$ *vzhledem k formuli* $\psi$. Jedná se o formuli $\mathsf{wlp}(\psi \mid p)$, která neobsahuje modality a která intuitivně vyjadřuje nejslabší tvrzení, které můžeme před provedením programu $p$ vyslovit, abychom po ukončení programu $p$ měli zaručenu pravdivost formule $\psi$. Ukážeme, že nejslabší liberální prepodmínku lze nalézt algoritmicky, pokud program $p$ neobsahuje žádný `while` cyklus. Nemožnost nalézt nejslabší liberální prepodmínku pro obecný `while` cyklus je jednou z motivací pro zavedení modálního $\mu$-kalkulu v kapitole 8.

**Kapitola 7** Tato kapitola mírně vybočuje z hlavního proudu textu. Pojem *modální algebry typu* $\Delta$, který je zde zaveden slouží jako nástroj pro důkaz věty o úplnosti systému **K** minimální modální logiky z kapitoly 5. Dokončení důkazu věty o úplnosti systému **K** je hlavním výsledkem této kapitoly.

Modální algebra $\mathbb{B}$ je Booleova algebra, která navíc obsahuje unární operaci $[\![R]\!]$ pro každé jméno akce $R$. Takto definovaná algebraická struktura dovoluje zavést pojem *ohodnocení* i pro modální logiku. Je samozřejmě zapotřebí ukázat, že pravdivost modální formule ve smyslu ohodnocení vyjde nastejno jako pravdivost ve smyslu sémantiky možných světů. Tento výsledek pak umožňuje větu o úplnosti systému **K** ukázat. Čtenář, který nechce tuto část studovat, tak může naprostou většinu kapitoly 7 přeskočit. Pojem modální algebry však je použit v kapitolách dalších.

**Kapitola 8** Zavedení modálního $\mu$-kalkulu je motivováno naší neschopností popsat nejslabší liberální prepodmínku pro obecný cyklus `while` jazykem modální logiky. Snaha najít nejslabší liberání prepodmínku pro cyklus `while` přirozeně vede na studium pevných bodů monotónních zobrazení.

V konkrétním modálním modelu $\mathscr{M}$ lze totiž na formuli $\varphi$ modální logiky nahlížet jako na množinu vrcholů $\|\varphi\|_{\mathscr{M}}$ akční struktury, ve kterých je formule $\varphi$ pravdivá. Jednotlivé logické spojky pak lze chápat jako operace s takovými množinami. Tak například na spojku $\wedge$ lze nahlížet jako na operaci průniku, protože platí

$$\|\varphi \wedge \psi\|_{\mathscr{M}} = \|\varphi\|_{\mathscr{M}} \cap \|\psi\|_{\mathscr{M}}.$$

Takový pohled na formule byl již uplatněn v kapitole 7 při definici modální algebry, v kapitole 8 se objevuje nový aspekt:

> *Řadu požadavků, které jsme nemohli vyjádřit jazykem modální logiky, lze vyjádřit v řeči množin vrcholů akční struktury.*

Formule modálního $\mu$-kalkulu jsou tedy interpretovány jako množiny vrcholů akční struktury. Syntaxe těchto formulí je nutně složitější než syntaxe formulí modální logiky.

Ve formulích modálního $\mu$-kalkulu se mohou objevovat *proměnné*. Formule modálního $\mu$-kalkulu, která obsahuje *volnou proměnnou*, například formule $x$ je pak interpretována jako *monotónní funkce* jednoho argumentu na množině všech podmnožin množiny vrcholů akční struktury. V kapitole 8 ukážeme, že každá taková monotónní funkce má *nejmenší* a *největší pevný bod*. Protože pevné body monotónního zobrazení $f$ lze chápat jako řešení *rekursivní rovnice* $X = f(X)$ pro množinu vrcholů $X$, lze chápat formule modálního $\mu$-kalkulu jako řešení *rekursivních rovnic pro formule modální logiky*.

Modální $\mu$-kalkulus nám tak dává velmi bohatý jazyk, kterým lze hovořit o *temporálních vlastnostech* procesů. Jako příklady takových temporálních vlastností uvedeme vlastnosti Liveness a Safety:

Liveness: proces jednou přejde do „dobrého“ stavu.

Safety: proces nikdy nepřejde do „špatného“ stavu.

**Kapitola 9** Tato kapitola je věnována formálnímu zavedení *jazyka CCS*, kterým lze zadávat akční struktury. Jazyk CCS navíc umožňuje algebraický popis situace, kdy dva procesy spolu komunikují. Termy jazyka CCS lze chápat jako jednotlivé procesy. Tyto procesy jsou interpretovány ve speciální akční struktuře $\mathbb{CCS}$. Struktura $\mathbb{CCS}$ je syntaktickým modelem jazyka CCS a dovoluje nám definovat důležitý pojem *vývoje procesu*. Vývojem procesu $t$ tu rozumíme seznam všech možných budoucích procesů, do kterých se proces $t$ může vyvinout.

Dva procesy, které jsou různé na úrovni syntaxe, mohou být ztotožněny, pokud nás zajímá jejich vývoj. Tomuto ztotožnění říkáme *silná bisimilarita procesů*. Při důkazu toho, že silnou bisimilaritu lze popsat jako speciální zig-zag relaci — jde o *největší zig-zag relaci na akční struktuře* $\mathbb{CCS}$ — se budeme opírat o teorii pevných bodů monotónních zobrazení.

Pojem silné bisimilarity tak umožňuje čistě algebraickou manipulaci s termy jazyka CCS. V kapitole 9 uvedeme řadu pravidel, která nám dovolí definice procesů v jazyce CCS zjednodušit.

# Kapitola 2

# Rychlé repetitorium

V této kapitole zopakujeme některé pojmy, které budeme dále používat. Dodržujeme standardní množinové značení, tj.

$x \in A$ značí, že $x$ je prvkem množiny $A$,

je-li $A$ množina a $V$ „vlastnost“ [1], pak $\{x \in A \mid V\}$ značí množinu všech $x \in A$ takových, že $x$ má vlastnost $V$,

$\emptyset$ značí prázdnou množinu,

$\subseteq$, $\cap$, $\cup$, $\setminus$, $\times$ značí po řadě inklusi, průnik, sjednocení, rozdíl množin, kartézský součin množin,

$exp(A)$ značí množinu všech podmnožin množiny $A$, tj. $exp(A) = \{M \mid M \subseteq A\}$,

$f : A \to B$ značí zobrazení (jiný termín: funkce) z množiny $A$ do množiny $B$ (definované ve všech bodech množiny $A$), hodnotu $f$ pro $x \in A$ značíme $f(x)$,

$\circ$ značí skládání zobrazení,

$1_A : A \to A$ značí identické zobrazení na množině $A$, tj. $1_A(x) = x$ pro všechna $x \in A$,

relace $C$ z množiny $A$ do množiny $B$ je nějaká množina uspořádaných dvojic $(a, b)$ tak, že $a \in A$ a $b \in B$; tj. $C \subseteq A \times B$, fakt, že $C$ je relace z $A$ do $B$ značíme $C : A \multimap B$,

$\mathbb{N}$ značí množinu všech přirozených čísel $\{0, 1, 2, \ldots\}$,

Zobrazení $f : A \to B$ je:

*prosté* (injektivní), pokud platí: jestliže $x \neq y$, pak $f(x) \neq f(y)$ (pro všechna $x, y \in A$),

*na* (surjektivní), pokud platí: pro libovolné $y \in B$ existuje $x \in A$ takové, že $f(x) = y$,

*vzájemně jednoznačné* (bijektivní), pokud je prosté a na současně.

Dvě množiny $A$ a $B$ mají *stejnou mohutnost* (stejný počet prvků), pokud existuje vzájemně jednoznačné zobrazení $f : A \to B$. Množina $A$ je *konečná*, pokud existuje $n \in \mathbb{N}$ takové, že $A$ má stejnou mohutnost jako množina $\{0, 1, \ldots, n-1\}$ (kde v případě $n = 0$ klademe $\{0, 1, \ldots, n-1\} = \emptyset$). Množina je *spočetná*, pokud má stejnou mohutnost jako množina všech přirozených čísel.

[1] Upozorňujeme, že přistupujeme k množinám *naivně*, ve smyslu kapitoly 1 [DePo97].

## 2.1 Binární relace a grafy

*Binární relace $R$ na množině $A$* je libovolná podmnožina kartézského součinu $A$ s $A$, tedy $R \subseteq A \times A$. Binární relaci je tedy možno chápat staticky jako seznam všech uspořádaných dvojic $(x, y)$ prvků z množiny $A$ takových, že $x$ má s $y$ „vztah“ $R$. Namísto $(x, y) \in R$ píšeme často obvyklejší infixní zápis $x\ R\ y$.

Každou uspořádanou dvojici si můžeme představit dynamicky jako „přechod“ od první složky ke druhé. Tato představa vede přirozeně k pojmu orientovaného grafu. Dvojici $(x, y) \in R$ si lze představit jako *orientovanou hranu* $x \to y$ vedoucí $x$ z do $y$. Definujeme tedy *orientovaný graf* jako uspořádanou dvojici $\langle A, R\rangle$, kde $A$ je množina a $R$ je binární relace na $A$. V tomto kontextu prvkům množiny $A$ říkáme *vrcholy*. V dalším budeme binární relace na množině a orientované grafy ztotožňovat. Dále, protože o jiných než orientovaných grafech nebudeme mluvit, budeme přídavné jméno *orientovaný* vynechávat.

Kdykoli definujeme nějakou matematickou strukturu (zde například grafy), zajímá nás možnost různé struktury mezi sebou porovnávat. Nástrojem pro porovnání struktur stejného typu bývají obvykle zobrazení, která v nějakém smyslu „respektují“ danou strukturu. Taková zobrazení chceme přirozeným způsobem skládat a chceme, aby každou strukturu šlo porovnat samu se sebou prostřednictvím identického zobrazení. Oblast matematiky, která studuje struktury a zobrazení mezi nimi na abstraktní úrovni, se jmenuje *teorie kategorií* — zájemce odkazujeme například na knihu [Bo94]. My teorii kategorií v tomto textu nebudeme nijak rozvíjet, přesto uvidíme, že situace *struktura a morfismy struktur*, bude pro naše zkoumání velmi důležitá. Situaci budeme ilustrovat na známém příkladu grafů a jejich homomorfismů.

Zobrazení, která „respektují“ strukturu grafu, jsou homomorfismy grafů.

**2.1.1 Definice** Ať $\langle A, R\rangle$ a $\langle B, S\rangle$ jsou grafy. Zobrazení $f : A \to B$ je *homomorfismus z* $\langle A, R\rangle$ *do* $\langle B, S\rangle$, (značíme $f : \langle A, R\rangle \to \langle B, S\rangle$) pokud platí:

jestliže $(x, y) \in R$, potom $(f(x), f(y)) \in S$ (pro libovolná $x, y \in A$).

Je evidentní, že složením homomorfismů vznikne opět homomorfismus. Dále je zřejmé, že identické zobrazení $1_A$ na libovolné množině $A$ je vždy homomorfismus (pro libovolný graf na množině $A$).[2]

Pojem homomorfismu grafů umožňuje některé grafy prohlásit za „vpodstatě stejné“, tj. definovat isomorfismus grafů.

**2.1.2 Definice** Řekneme, že homomorfismus $f : \langle A, R\rangle \to \langle B, S\rangle$ je *isomorfismus*, pokud existuje homorfismus $g : \langle B, S\rangle \to \langle A, R\rangle$ takový, že platí $g \circ f = 1_A$ a $f \circ g = 1_B$.

Grafy $\langle A, R\rangle$ a $\langle B, S\rangle$ jsou *isomorfní*, pokud existuje isomorfismus $f : \langle A, R\rangle \to \langle B, S\rangle$.

Homomorfismy neslouží pouze k porovnávání struktur, ale i vytváření nových struktur ze struktur již existujících.

Připomeňme dvě základní konstrukce nových grafů: *podgraf* a *součin grafů*.

**2.1.3 Definice** Řekneme, že $\langle B, S\rangle$ je *podgraf* grafu $\langle A, R\rangle$, pokud $B \subseteq A$ a platí, že pokud $(x, y) \in S$, pak $(x, y) \in R$ (pro libovolná $x, y \in B$).

$\langle B, S\rangle$ je *úplný podgraf* grafu $\langle A, R\rangle$, pokud $B \subseteq A$ a platí, že $(x, y) \in S$ právě tehdy, když $(x, y) \in R$ (pro libovolná $x, y \in B$).

Jsou-li dány grafy $\langle A_i, R_i\rangle$, kde $i \in I$, pak jejich *kartézským součinem* rozumíme graf $\langle A, R\rangle$, kde $A = \prod_{i\in I} A_i$ (kartézský součin systému množin $A_i$) a dvojice $(x, y)$ je v relaci $R$ právě tehdy, když platí

$(x_i, y_i) \in R_i$ pro každé $i \in I$,

kde např. $x_i$ značí $i$-tou souřadnici $x$. Kartézský součin značíme $\prod_{i\in I} \langle A_i, R_i\rangle$.

Kartézský součin grafů je konstrukce, definovaná „až na isomorfismus“. Přesněji, platí následující dvě tvrzení:

**2.1.4 Lemma** *Ať je dán systém grafů $\langle A_i, R_i\rangle$, $i \in I$ a ať $\langle A, R\rangle = \prod_{i\in I} \langle A_i, R_i\rangle$ je jejich kartézský součin.*

[2] V jazyce teorie kategorií bychom zde hovořili o *kategorii grafů a jejich homomorfismů*.

1. *Pak, pro libovolné $i \in I$, je projekce $p^i : \prod_{i\in I} A_i \to A_i$ homomorfismem grafů.*

   *Navíc, je-li dán graf $\langle X, S\rangle$ a systém homomorfismů $f^i : \langle X, S\rangle \to \langle A_i, R_i\rangle$, $i \in I$, pak existuje právě jeden homomorfismus $f : \langle X, S\rangle \to \langle A, R\rangle$ takový, že $p^i \circ f = f^i$ pro všechna $i \in I$.*

2. *Je-li $\langle G, P\rangle$ graf a je-li $g^i : \langle G, P\rangle \to \langle A_i, R_i\rangle$, $i \in I$, a systém homomorfismů s následující vlastností:*

   *Pro libovolný graf $\langle X, S\rangle$ a systém homomorfismů $f^i : \langle X, S\rangle \to \langle A_i, R_i\rangle$, $i \in I$, existuje právě jeden homomorfismus $f : \langle X, S\rangle \to \langle G, P\rangle$ takový, že $g^i \circ f = f^i$ pro všechna $i \in I$.*

   *pak graf $\langle G, P\rangle$ je isomorfní součinu $\prod_{i\in I} \langle A_i, R_i\rangle$*

Důkaz. Viz cvičení 2.2.5. ■

Podobně prostý homomorfismus grafů lze chápat jako konstrukci podgrafu, definovanou „až na isomorfismus“. Přesněji, platí následující tvrzení:

**2.1.5 Lemma** *Ať $\langle A, R\rangle$ je graf. Potom platí:*

1. *Je-li $\langle B, S\rangle$ je podgraf grafu $\langle A, R\rangle$, pak zobrazení $f : B \to A$, definované předpisem $f(b) = b$ pro všechna $b \in B$, je* prostý homomorfismus grafů. *(Toto tvrzení se často zkráceně vyjadřuje:* inkluse podgrafu je homomorfismus.*)*

2. *Je-li $f : \langle B, S\rangle \to \langle A, R\rangle$ prostý homomorfismus grafů, pak existuje podgraf $\langle B', S'\rangle$ grafu $\langle A, R\rangle$ takový, že grafy $\langle B', S'\rangle$ a $\langle B, S\rangle$ jsou isomorfní.*

Důkaz. Viz cvičení 2.2.6. ■

Lemma 2.1.4 lze využít jako *novou* definici součinu systému grafů:

*Součinem systém grafů $\langle A_i, R_i\rangle$, $i \in I$* rozumíme graf $\langle A, R\rangle$ a homomorfismy $p^i : \langle A, R\rangle \to \langle A_i, R_i\rangle$, $i \in I$ s následující vlastností:

Je-li dán libovolný graf $\langle X, S\rangle$ a libovolný systém homomorfismů $f^i : \langle X, S\rangle \to \langle A_i, R_i\rangle$, $i \in I$, pak existuje právě jeden homomorfismus $f : \langle X, S\rangle \to \langle A, R\rangle$ takový, že platí $p^i \circ f = f^i$ pro všechna $i \in I$.

Podobné definice jsou typické v teorii kategorií: zaváděný pojem je definován pouze pomocí vlastností homomorfismů. Takové definice mají tu výhodu, že umožňují okamžitě definovat i *duální pojem*. Předvedeme definici duálního pojmu k součinu grafů:

**2.1.6 Definice** *Koproduktem systém grafů $\langle A_i, R_i\rangle$, $i \in I$* rozumíme graf $\langle A, R\rangle$ (značený $\coprod_{i\in I} \langle A_i, R_i\rangle$) a homomorfismy $p^i : \langle A_i, R_i\rangle \to \langle A, R\rangle$, $i \in I$ s následující vlastností:

Je-li dán libovolný graf $\langle X, S\rangle$ a libovolný systém homomorfismů $f^i : \langle A_i, R_i\rangle \to \langle X, S\rangle$, $i \in I$, pak existuje právě jeden homomorfismus $f : \langle A, R\rangle \to \langle X, S\rangle$ takový, že platí $f \circ p^i = f^i$ pro všechna $i \in I$.

Vzniká samozřejmě otázka, zda každý systém grafů má koprodukt. Ukážeme, že ano a tento koprodukt popíšeme.

**2.1.7 Lemma** *Ať $\langle A_i, R_i\rangle$, $i \in I$ je systém grafů. Definujme graf $\langle A, R\rangle$ a homomorfismy $p^i : \langle A_i, R_i\rangle \to \langle A, R\rangle$ následovně:*

1. *Množina $A$ je disjunktní sjednocení množin $A_i$. Přesněji, ze systému množin $A_i$ uděláme nejprve systém množin $A'_i$ tak, aby množiny $A'_i$ byly navzájem disjunktní: definujeme $A'_i = A_i \times \{i\}$. Položíme $A = \bigcup_{i\in I} A'_i$.*

2. *Relace $R$ na množině $A$ je definována následovně: prvky $x$ a $y$ z $A$ jsou v relaci $R$, pokud platí:*

*existuje $i \in I$ tak, že $x$ je tvaru $(a, i)$ pro nějaké $a \in A_i$ a $y$ je tvaru $(b, i)$ pro nějaké $b \in A_i$ a platí $a\ R_i\ b$.*

3. *Pro $i \in I$ je homomorfismus $p^i : \langle A_i, R_i\rangle \to \langle A, R\rangle$ definován jako zobrazení $a \mapsto (a, i)$.*

Důkaz. Viz cvičení 2.2.7. ■

Konstrukce koproduktu grafů je velmi názorná: vezměte grafy $\langle A_i, R_i\rangle$ a „položte je vedle sebe tak, aby se nepřekrývaly“ — dostanete tak nový graf $\langle A, R\rangle$. To je intuitivní význam definice množiny $A$ a relace $R$ v předchozím lemmatu. Homomorfismus grafů $p^i$ je pak zobrazení, které „vkládá“ graf $\langle A_i, R_i\rangle$ do grafu $\langle A, R\rangle$.

V dalším textu budeme používat obecnější pojem grafu — akční strukturu (viz definice 3.1.1). Pojmy a výsledky pro akční struktury samozřejmě lze využít i pro studium grafů. Nebudeme tedy v této kapitole další pojmy pro grafy zavádět.

Studium grafů je vlastně také studium binární relace na množině. Připomeňme, že binární relace na množině mohou mít řadu dodatečných vlastností. Zmiňme některé z nich:

**2.1.8 Definice** Relace $R$ na množině $A$ je:

1. *reflexivní*, pokud $x\ R\ x$ platí pro všechna $x \in A$.
2. *symetrická*, pokud pro všechna $x, y \in A$ platí: je-li $x\ R\ y$, pak $y\ R\ x$.
3. *transitivní*, pokud pro všechna $x, y, z \in A$ platí: je-li $x\ R\ y$ a současně $y\ R\ z$, pak $x\ R\ z$.
4. *antisymetrická*, pokud pro všechna $x, y \in A$ platí: je-li $x\ R\ y$ a současně $y\ R\ x$, pak $x = y$.

Nakombinováním těchto jednoduchých vlastností dostáváme vlastnosti složitější:

**2.1.9 Definice** Relace $R$ na množině $A$ je:

1. *kvasiuspořádání*, pokud je reflexivní a transitivní.
2. *částečné uspořádání*, pokud je reflexivní, transitivní a antisymetrická.
3. *similarita*, pokud je reflexivní a symetrická.
4. *ekvivalence*, pokud je reflexivní, symetrická a transitivní.

Připomeňme, že ekvivalence $R$ na neprázdné množině $A$ dává vzniknout *rozkladu $A$ podle $R$* následujícím způsobem:

1. Prvky rozkladu $A$ podle $R$ jsou *třídy ekvivalence*. Třída ekvivalence prvku $x \in A$ je množina $[x]_R = \{y \in A \mid x\ R\ y\}$. Každá třída ekvivalence je neprázdná množina.
2. Systém množin $[x]_R = \{y \in A \mid x\ R\ y\}$ tvoří rozklad $A$ v následujícím smyslu: pro $x, x' \in A$ platí buď $[x]_R = [x']_R$ nebo $[x]_R \cap [x']_R = \emptyset$.

Obecně *rozkladem neprázdné množiny $A$* rozumíme systém neprázdných množin $A_i$, $i \in I$ s následujícími vlastnostmi:

1. Každá množina $A_i$ je podmnožinou $A$ a sjednocení všech množin $A_i$ je množina $A$.
2. Jestliže $A_i \cap A_j \neq \emptyset$, potom $i = j$.

Třídy ekvivalence $[x]_R$ zřejmě tvoří rozklad množiny $A$ podle výše uvedené definice. Navíc, každý rozklad množiny $A$ lze chápat jako rozklad podle nějaké relace ekvivalence $R$. Zformulujme to přesně:

**2.1.10 Lemma** *Ať $A$ je neprázdná množina. Potom platí:*

1. *Každá relace ekvivalence $R$ určuje rozklad $A/R = \{[x]_R \mid x \in A\}$.*
2. *Je-li dán rozklad $\mathcal{A} = \{A_i \mid i \in I\}$ množiny $A$, definujme binární relaci $R_{\mathcal{A}}$ na $A$ následovně:*

*$x$ $R_{\mathcal{A}}$ $y$ platí právě tehdy, když existuje $i \in I$ tak, že $x \in A_i$ a $y \in A_i$.*

*Potom $R_{\mathcal{A}}$ je relace ekvivalence na $A$.*

*Navíc platí: zobrazení*

$$R \mapsto A/R \mapsto R_{A/R} \quad \text{a} \quad \mathcal{A} \mapsto R_{\mathcal{A}} \mapsto A/R_{\mathcal{A}}$$

*jsou identity.*

Důkaz. Viz cvičení 2.2.8. ■

Rozklady (ekvivalence) na množině $A$ lze charakterisovat i pomocí surjektivních zobrazení vedoucích z množiny $A$.

**2.1.11 Lemma** *Ať $A$ je neprázdná množina. Potom platí:*

1. *Pro relaci ekvivalence $R$ na $A$ je zobrazení $\pi_R : A \to A/R$ definované jako $x \mapsto [x]_R$ surjektivní.*
2. *Je-li $f : A \to B$ surjektivní zobrazení, potom binární relace $\ker f$ definovaná předpisem*

   *$x \ker f$ $y$ platí právě tehdy, když $f(x) = f(y)$*

   *je relace ekvivalence na množině $A$.*

*Navíc platí: zobrazení*

$$R \mapsto \pi_R \mapsto \ker \pi_R$$

*je identita.*

Důkaz. Viz cvičení 2.2.8. ■

## 2.2 Cvičení

**2.2.1 Cvičení** Ukažte, že pro zobrazení $f : A \to B$ platí:

1. $f$ je prosté, právě tehdy, když platí: jestliže $f \circ u = f \circ v$, pak $u = v$ (pro libovolná $u$, $v$).
2. $f$ je surjektivní, právě tehdy, když platí: jestliže $u \circ f = v \circ f$, pak $u = v$ (pro libovolná $u$, $v$).
3. $f$ je vzájemně jednoznačné, právě tehdy, když platí: existuje $g : B \to A$ takové, že $g \circ f = 1_A$ a $f \circ g = 1_B$.

**2.2.2 Cvičení** Ať $R$ je symetrická a transitivní relace na množině $A$. Je potom relace $R$ reflexivní?

**2.2.3 Cvičení** Relace $R$ na množině $A$ je:

1. *seriální*, pokud pro každé $x \in A$ existuje $y \in A$ takové, že platí $x \; R \; y$.
2. *eukleidovská*, pokud pro každé $x, y, z \in A$ platí: je-li $x \; R \; y$ a současně $x \; R \; z$, pak $y \; R \; z$.

Ukažte, že pro relaci $R$ na množině $A$ jsou následující podmínky ekvivalentní:

1. Relace $R$ je ekvivalence.
2. Relace $R$ je reflexivní a eukleidovská.
3. Relace $R$ je seriální, symetrická a transitivní.
4. Relace $R$ je seriální, symetrická a eukleidovská.

**2.2.4 Cvičení** Přepokládejte, že $R$ je relace kvasiuspořádání na množině $A$. Binární relaci $E_R$ na $A$ definujte následovně:

$x\ E_R\ y$ platí právě tehdy, když platí $x\ R\ y$ a $y\ R\ x$ současně.

Ukažte, že relace $E_R$ je ekvivalence na množině $A$. Ukažte, že binární relace $[R]$ definovaná na množině všech tříd ekvivalence $E_R$ následujícím způsobem:

$[x]_{E_R}\ [R]\ [y]_{E_R}$ platí právě tehdy, když platí $x\ R\ y$,

je relace částečného uspořádání.

**2.2.5 Cvičení** Dokažte lemma 2.1.4.

**2.2.6 Cvičení** Dokažte lemma 2.1.5.

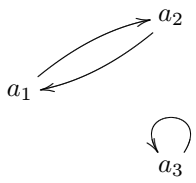
**2.2.7 Cvičení** Dokažte lemma 2.1.7.

**2.2.8 Cvičení** Dokažte lemmata 2.1.10 a 2.1.11.

# Kapitola 3

# Procesy a akční struktury

V této kapitole definujeme základní pojem, se kterým budeme nadále pracovat; *akční strukturu*. Jedná se o jednoduché zobecnění orientovaného grafu. Dynamický pohled na orientovaný graf nám umožnil chápat hranu $x \to y$ v grafu $\langle A, R\rangle$ jako *přechod ze stavu $x$ do stavu $y$*. Následující obrázek grafu




lze tedy chápat takto:

1. Množina stavů je množina $A = \{a_1, a_2, a_3\}$.

2. Binární relace $R = \{(a_1, a_2), (a_2, a_1), (a_3, a_3)\}$ je chápána jako *výčet všech dvojic stavů $(x, y)$ takových, že ze stavu $x$ lze přejít do stavu $y$.*

Tuto myšlenku zobecňuje pojem *akční struktury*: dovolíme, aby přechody mezi stavy měly různá jména. To znamená, že na jedné množině budeme uvažovat více binárních relací současně.

Po zavedení pojmu akční struktury zavedeme v části 3.3 i pojem morfismu akčních struktur. Výsledné pojmy *zig-zag morfismu* a jeho relačního zobecnění *zig-zag relace* budou pro nás později velmi důležité při studiu logiky. Zig-zag morfismy nám navíc pro akční struktury umožní definovat pojem *podstruktury* a *faktorové struktury*.

## 3.1 Akční struktury

**3.1.1 Definice** Ať je dána množina (může být i prázdná) $\Delta$. Prvkům množiny $\Delta$ budeme říkat *jména akcí* a typický prvek množiny $\Delta$ budeme značit $R$. Množině $\Delta$ pak budeme říkat *typ*.

*Akční struktura typu* $\Delta$ je dvojice $\mathbb{V} = \langle V, \{[\![R]\!]_{\mathbb{V}} \mid R \in \Delta\}\rangle$, kde:

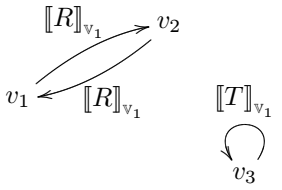
1. $V$ je *množina stavů* akční struktury $\mathbb{V}$, typický prvek $V$ budeme značit $v$.

2. Pro každé $R \in \Delta$ je $[\![R]\!]_{\mathbb{V}}$ binární relace na množině $V$. Jedná se o *interpretaci jména akce $R$ v množině $V$*.

Akční struktura je tedy „více grafů na jedné množině současně“. Typ akční struktury je seznam jmen akcí, které jsou pak v konkrétní akční struktuře daného typu interpretovány jako binární relace.

**3.1.2 Příklady** Uvedeme nyní několik příkladů akčních struktur. Ať typ $\Delta$ obsahuje pouze prvky $R, S, T$.

1. Následující obrázek je pak příkladem akční struktury typu $\Delta$:




   Množina stavů je $V_1 = \{v_1, v_2, v_3\}$, interpretací jména $R$ je binární relace $[\![R]\!]_{\mathbb{V}_1} = \{(v_1, v_2), (v_2, v_1)\}$, interpretací jména $T$ je binární relace $[\![T]\!]_{\mathbb{V}_1} = \{(v_3, v_3)\}$ a interpretací jména $S$ je binární relace $[\![S]\!]_{\mathbb{V}_1} = \emptyset$.

2. Jiným příkladem je množina $V_2 = \{v\}$, kde všechna jména z množiny $\Delta$ jsou interpretována jako prázdné relace, tj. $[\![R]\!]_{\mathbb{V}_2} = [\![S]\!]_{\mathbb{V}_2} = [\![T]\!]_{\mathbb{V}_2} = \emptyset$. Obrázek takové akční struktury je pak




3. Definice akční struktury ovšem připouští, aby i množina vrcholů byla prázdná. Jména akcí však potom nelze interpretovat jinak, než jako prázdné relace.

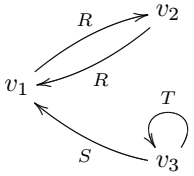
**3.1.3 Konvence** V dalším textu naše přísné rozlišování jména akce $R$ a interpretace $[\![R]\!]_{\mathbb{V}}$ v akční struktuře $\mathbb{V}$ nebudeme dodržovat, pokud to nepovede k nedorozumění. Budeme tedy psát $R$ jak pro jméno akce, tak pro interpretaci jména.

Dále budeme namísto $(v, v') \in R$ psát $v \xrightarrow{R} v'$.

## 3.2 Zadávání akčních struktur

Zadávání akčních struktur obrázkem nebo popisem akcí jako v příkladech 3.1.2 nemusí být vždy efektivní. Robin Milner ([Mi89]) zavedl notaci zvanou *CCS* (*Calculus of Communicating Systems*), která je vhodná pro zadávání akčních struktur. V této části uvedeme pouze fragment CCS. Další konstrukce kalkulu CCS ukážeme v kapitole 9.

**3.2.1 Příklad** Uvažujme o akční struktuře $\mathbb{V}$ typu $\{R, S, T\}$ dané následujícím obrázkem.




V jazyce CCS by definice $\mathbb{V}$ vypadala takto:

$$v_1 \stackrel{\text{def}}{=} R.v_2 \tag{3.1}$$

$$v_2 \stackrel{\text{def}}{=} R.v_1 \tag{3.2}$$

$$v_3 \stackrel{\text{def}}{=} T.v_3 + S.v_1 \tag{3.3}$$

Výše uvedená notace definuje tři stavy: $v_1$, $v_2$ a $v_3$.

Rovnice (3.1) definuje stav $v_1$ jako stav, ze kterého je možné vykonat akci $R$ a skončit tak ve stavu $v_2$.

Stav $v_2$ je stav, ze kterého je možné vykonat akci $R$ a skončit tak ve stavu $v_1$ — to říká rovnice (3.2).

Konečně smyslem rovnice (3.3) je definice stavu $v_3$, ze kterého je možné vykonat akci $T$ a skončit tak ve stavu $v_3$ *nebo* je možné vykonat akci $S$ a skončit tak ve stavu $v_1$.

Fragment jazyka CCS, který budeme zatím používat, je dán následovně:

1. *Term jazyka CCS* je buď $v$ nebo $R.v$, kde $v$ je prvek nějaké množiny $V$ a $R$ je prvek množiny $\Delta$.

2. *Fráze jazyka CCS* je zápis

$$v \stackrel{\text{def}}{=} R.w$$

(čteme: $v$ je definováno jako $R$ tečka $w$) nebo

$$v \stackrel{\text{def}}{=} R_1.v_1 + R_2.v_2$$

(čteme: $v$ je definováno jako $R_1$ tečka $v_1$ nebo $R_2$ tečka $v_2$) nebo

$$v \stackrel{\text{def}}{=} \sum_{i \in I} R_i.v_i,$$

kde $I$ je nějaká množina (čteme: $v$ je definováno jako suma $R_i$ tečka $v_i$ pro $i \in I$).

Souvislost frází jazyka CCS s akcemi je dána následujícími odvozovacími pravidly:

$$\frac{}{R.v \stackrel{R}{\to} v}\ (\text{prefix}), \qquad \text{je-li } R \in \Delta \tag{3.4}$$

$$\frac{w \stackrel{R}{\to} w'}{v \stackrel{R}{\to} w'}\ (\stackrel{\text{def}}{=}), \qquad \text{je-li } v \stackrel{\text{def}}{=} w \tag{3.5}$$

$$\frac{v_{i_0} \stackrel{R}{\to} v}{\sum_{i \in I} v_i \stackrel{R}{\to} v}\ (\textstyle\sum), \qquad \text{je-li } i_0 \in I \tag{3.6}$$

Odvozovací pravidlo (3.4) je třeba číst jako axiom: na to, abychom odvodili akci $R.v \stackrel{R}{\to} v$, potřebujeme vědět pouze, že $R \in \Delta$.

Pravidlo (3.5) je odvozovací pravidlo: víme-li, že $v \stackrel{\text{def}}{=} w$ a víme-li, že $w \stackrel{R}{\to} w'$, pak odvodíme akci $v \stackrel{R}{\to} w'$.

Pravidlo (3.6) je také chápáno jako odvozovací pravidlo: víme-li, že $i_0 \in I$ a víme-li, že $v_{i_0} \stackrel{R}{\to} v$, pak odvodíme akci $\sum_{i \in I} v_i \stackrel{R}{\to} v$.

**3.2.2 Příklad** Ukážeme, jak pomocí pravidel odvodíme akci $v_3 \stackrel{S}{\to} v_1$ v akční struktuře z příkladu 3.2.1. Existence této akce je potvrzena konstrukcí *derivačního stromu* této akce pomocí pravidel jazyka CCS:

$$\cfrac{\cfrac{\cfrac{}{S.v_1 \stackrel{S}{\to} v_1}\ (\text{prefix})}{T.v_3 + S.v_1 \stackrel{S}{\to} v_1}\ (\textstyle\sum)}{v_3 \stackrel{S}{\to} v_1}\ (\stackrel{\text{def}}{=})$$

Derivační strom čteme odshora dolů:

1. Axiom (prefix) nám umožnil odvodit akci $S.v_1 \stackrel{S}{\to} v_1$.

2. Z akce $S.v_1 \stackrel{S}{\to} v_1$ jsme pravidlem $(\sum)$ odvodili akci $T.v_3 + S.v_1 \stackrel{S}{\to} v_1$.

3. Akci $v_3 \stackrel{S}{\to} v_1$ jsme z akce $T.v_3 + S.v_1 \stackrel{S}{\to} v_1$ odvodili pravidlem $(\stackrel{\text{def}}{=})$, protože $v_3 \stackrel{\text{def}}{=} T.v_3 + S.v_1$.

Pro další příklady vztahu CCS a akčních struktur odkazujeme na cvičení 3.4.2, 3.4.3 a 3.4.4. Fragment jazyka CCS dále rozšíříme v kapitole 9.

## 3.3 Morfismy akčních struktur

Později budeme chtít akční struktury stejného typu porovnávat mezi sebou. Pro tyto účely je vhodný pojem morfismu akčních struktur. První nápad je nasnadě: akční struktura je „více grafů na jedné množině“, morfismem by tedy mohlo být zobrazení, které respektuje strukturu všech těchto grafů.

**3.3.1 Definice** Ať $\mathbb{V} = \langle V, \{[\![R]\!]_{\mathbb{V}} \mid R \in \Delta\}\rangle$ a $\mathbb{W} = \langle W, \{[\![R]\!]_{\mathbb{W}} \mid R \in \Delta\}\rangle$ jsou akční struktury typu $\Delta$. Zobrazení $f : V \to W$ takové, že pro všechna $R \in \Delta$ platí

jestliže $x \xrightarrow{R} y$ ve $\mathbb{V}$, pak $f(x) \xrightarrow{R} f(y)$ ve $\mathbb{W}$

nazveme *silný morfismus z* $\mathbb{V}$ *do* $\mathbb{W}$.

Jak uvidíme, není tento pojem morfismu pro naše účely vhodný. Přesto o pojmu silného morfismu učiňme následující základní pozorování. Důkaz je jednoduchý a přenecháváme jej jako cvičení.

**3.3.2 Lemma** *Identické zobrazení je vždy silný morfismus. Složením dvou silných morfismů je opět silný morfismus.*

Nevýhody silného morfismu pochopíme, když si uvědomíme vztah mezi funkcemi (z množiny $A$ do množiny $B$) a relacemi (z množiny $A$ do množiny $B$). Připomeňme, že $C : A \multimap B$ je *relace z množiny $A$ do množiny $B$*, když $C \subseteq A \times B$. Samozřejmě, každá funkce $f : A \to B$ je příkladem relace z množiny $A$ do množiny $B$. Funkce $f$ je totiž taková relace $f : A \multimap B$, kde pro každé $x \in A$ existuje právě jedno $y \in B$ takové, že $(x, y) \in f$. Toto jediné $y$ příslušející $x$ značíme $f(x)$. Existují ovšem relace, které nejsou funkcemi. Prvním pozorováním o relacích a funkcích tedy je:

1 Relace zobecňují funkce.

Dále pro relaci $C : A \multimap B$ můžeme uvažovat o opačné (jiný termín je inversní) relaci $C^{op} : B \multimap A$. V relaci $C^{op}$ jsou přesně ty dvojice $(y, x)$, kde $(x, y) \in C$. Druhé pozorování o relacích a funkcích je následující:

2 Je-li $f$ funkce z $A$ do $B$, pak $f^{op}$ již nemusí být funkce z $B$ do $A$ ($f^{op}$ je však samozřejmě relace z $B$ do $A$).

Jaké by tedy mělo být zobecnění silného morfismu z akční struktury $\mathbb{V}$ do akční struktury $\mathbb{W}$ na „silnou relaci“ (z akční struktury $\mathbb{V}$ do akční struktury $\mathbb{W}$)?

Jednalo by se zřejmě o relaci $C : V \multimap W$ takovou, že pro každé jméno akce $R$ je splněna následující podmínka:

(F) je-li $v\ C\ w$ a současně $v \xrightarrow{R} v'$ ve $\mathbb{V}$, pak existuje takové $w'$, pro které platí $v'\ C\ w'$ a současně $w \xrightarrow{R} w'$ ve $\mathbb{W}$.

Uvědomme si, že pak skutečně platí následující.

**3.3.3 Lemma** *Je-li $f$ silný morfismus z $\mathbb{V}$ do $\mathbb{W}$, pak relace $f$ splňuje podmínku* (F).

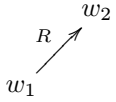
Důkaz. Předpokládáme-li, že $v\ f\ w$, znamená to, že $w = f(v)$. Máme-li ve $\mathbb{V}$ akci $v \xrightarrow{R} v'$, položme $w' = f(v')$. Potom samozřejmě platí $v'\ f\ w'$ a navíc $w \xrightarrow{R} w'$, protože $f$ je silný homomorfismus. ■

Relace splňující podmínku (F) tedy zobecňují pojem silného morfismu z funkcí na relace. Problémem však je, že existuje relace $C$, která splňuje podmínku (F), ovšem relace $C^{op}$ podmínku (F) nesplňuje.

**3.3.4 Příklad** Vezměme typ $\Delta = \{R\}$. Za akční strukturu $\mathbb{V}$ vezměme strukturu

$$v$$

a za akční strukturu $\mathbb{W}$ vezměme akční strukturu




Relace $C = \{(v, w_1), (v, w_2)\}$ zřejmě splňuje podmínku (F), ale relace $C^{op} = \{(w_1, v), (w_2, v)\}$ nikoli.

Zkusme tedy charakterisovat ty situace, kdy i relace $C^{op}$ splňuje podmínku (F). Je to jednoduché.

**3.3.5 Lemma** *Ať $C$ je relace splňující podmínku* (F). *Pak následující je ekvivalentní:*

1. *Relace $C^{op}$ splňuje podmínku* (F).
2. *Relace $C$ splňuje pro každé jméno akce $R$ následující podmínku:*

   (B) *je-li $v\ C\ w$ a současně $w \xrightarrow{R} w'$ ve $\mathbb{W}$, pak existuje takové $v'$, pro které platí $v'\ C\ w'$ a současně $v \xrightarrow{R} v'$ ve $\mathbb{V}$.*

Důkaz. Triviální. ■

Splňuje-li relace $C$ obě podmínky (F) a (B) současně, máme zajištěnu potřebnou symetrii.

**3.3.6 Lemma** *Pro relaci $C : \mathbb{V} \multimap \mathbb{W}$ je následující ekvivalentní:*

1. *Relace $C$ splňuje podmínky* (F) *a* (B).
2. *Relace $C^{op}$ splňuje podmínky* (F) *a* (B).

Důkaz. Triviální. ■

To nás vede k následující definici.

**3.3.7 Definice** Ať $\mathbb{V} = \langle V, \{[\![R]\!]_{\mathbb{V}} \mid R \in \Delta\}\rangle$ a $\mathbb{W} = \langle W, \{[\![R]\!]_{\mathbb{W}} \mid R \in \Delta\}\rangle$ jsou akční struktury typu $\Delta$. Relaci $C : V \multimap W$ takovou, že pro všechna $R \in \Delta$ platí následující dvě podmínky

(F) je-li $v\ C\ w$ a současně $v \xrightarrow{R} v'$ ve $\mathbb{V}$, pak existuje takové $w'$, pro které platí $v'\ C\ w'$ a současně $w \xrightarrow{R} w'$ ve $\mathbb{W}$.

(B) je-li $v\ C\ w$ a současně $w \xrightarrow{R} w'$ ve $\mathbb{W}$, pak existuje takové $v'$, pro které platí $v'\ C\ w'$ a současně $v \xrightarrow{R} v'$ ve $\mathbb{V}$.

nazveme *zig-zag relace z $\mathbb{V}$ do $\mathbb{W}$*. Zig-zag relaci značíme $C : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\circ\ \mathbb{W}$. Je-li relace $C$ funkce, mluvíme o *zig-zag morfismu*. Zig-zag morfismus značíme $C : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\rightarrow\ \mathbb{W}$.

**3.3.8 Poznámka** Vrcholy, které jsou v zig-zag relaci, si lze představit jako „procesy, které mají možnost simulovat stejný vývoj“.

Předpokládejme, že $\mathbb{V}$ a $\mathbb{W}$ jsou akční struktury stejného typu a že $C : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\circ\ \mathbb{W}$. Dále předpokládejme, že pro vrcholy $v$ a $w$ platí $v\ C\ w$. Platnost vztahu $v\ C\ w$ svědčí o „simulovaní totožného vývoje“ vrcholů $v$ a $w$ v následujícím smyslu:

> Jestliže $v \xrightarrow{R} v_1$, potom existuje $w \xrightarrow{R} w_1$ a platí $v_1\ C\ w_1$. To nám předepisuje vlastnost (F) z definice zig-zag relace. Pro jakoukoli činnost procesu $v$ akcí $R$ (a přechodu v proces $v_1$) je při procesu $w$ možné „zareagovat“ podobným chováním: akcí $w \xrightarrow{R} w_1$, navíc to lze udělat tak, že platí $v_1\ C\ w_1$. Tato podmínka tedy říká, že podobný vývoj mohou simulovat i procesy $v_1$ a $w_1$.
>
> Jestliže $w \xrightarrow{R} w_2$, potom existuje $v \xrightarrow{R} v_2$ a platí $v_2\ C\ w_2$. To nám předepisuje vlastnost (B) z definice zig-zag relace. Pro jakoukoli činnost procesu $w$ akcí $R$ (a přechodu v proces $w_2$) je při procesu $v$ možné „zareagovat“ podobným chováním: akcí $v \xrightarrow{R} v_2$, navíc to lze udělat tak, že platí $v_2\ C\ w_2$. Tato podmínka tedy říká, že podobný vývoj mohou simulovat i procesy $v_2$ a $w_2$.

Vývoji procesů se budeme podrobněji věnovat v kapitole 9.

Zig-zag morfismy lze charakterisovat následovně:

**3.3.9 Lemma** *Jsou-li $\mathbb{V}$ a $\mathbb{W}$ dvě akční struktury, pak pro zobrazení $f : V \to W$ jsou následující podmínky ekvivalentní:*

1. *$f : \mathbb{V} \to \mathbb{W}$ je zig-zag morfismus.*
2. *Zobrazení $f$ je silný homomorfismus splňující navíc podmínku* (B).

Důkaz. 1. $\Rightarrow$ 2. Stačí ukázat, že $f$ je silný homomorfismus. Předpokládejme, že $v \xrightarrow{R} v'$ a označme $w = f(v)$. Protože zobrazení $f$ splňuje podmínku (F), existuje $w'$ takové, že $w \xrightarrow{R} w'$ a navíc $w' = f(v')$. Tudíž $f$ je silný homomorfismus.

2. $\Rightarrow$ 1. Stačí ukázat, že zobrazení $f$ splňuje podmínku (F). Vezměme libovolnou akci $v \xrightarrow{R} v'$ ve $\mathbb{V}$. Protože $f$ je silný homomorfismus, máme ve $\mathbb{W}$ akci $f(v) \xrightarrow{R} f(v')$. To ale znamená, že $f$ splňuje podmínku (F), protože $v' \; f \; f(v')$. ■

Pro zig-zag relace platí následující:

**3.3.10 Lemma** *Identická relace je vždy zig-zag relace. Složením dvou zig-zag relací vzniká opět zig-zag relace.*

Důkaz. Cvičení 3.4.1. ■

Protože složením dvou funkcí (chápaných jako relace) vzniká opět funkce a protože identická relace a identické zobrazení je totéž, dostáváme jako důsledek:

**3.3.11 Důsledek** *Identické zobrazení je vždy zig-zag morfismus. Složením dvou zig-zag morfismů vzniká opět zig-zag morfismus.*

Protože zig-zag morfismy tvoří pojem „správného zobrazení“ mezi akčními strukturami, umožňují zig-zag morfismy porovnávat akční struktury mezi sebou podobně, jako homomorfismy grafů umožňují porovnávat grafy.

**3.3.12 Definice** Řekneme, že zig-zag morfismus $f : \mathbb{V} \nrightarrow \mathbb{W}$ je *zig-zag isomorfismus*, pokud existuje zig-zag morfismus $g : \mathbb{W} \nrightarrow \mathbb{V}$ takový, že $f \circ g = 1_{\mathbb{W}}$ a $g \circ f = 1_{\mathbb{V}}$.

Struktury $\mathbb{V}$ a $\mathbb{W}$ jsou *isomorfní*, pokud existuje zig-zag isomorfismus $f : \mathbb{V} \to \mathbb{W}$.

**3.3.13 Lemma** *Pro akční struktury $\mathbb{V}$ a $\mathbb{W}$ je následující ekvivalentní:*

1. *$\mathbb{V}$ a $\mathbb{W}$ jsou zig-zag isomorfní.*
2. *$\mathbb{V}$ a $\mathbb{W}$ jsou silně isomorfní, tj. existují silné homomorfismy $f : \mathbb{V} \to \mathbb{W}$ a $g : \mathbb{W} \to \mathbb{V}$ tak, že $f \circ g = 1_{\mathbb{W}}$ a $g \circ f = 1_{\mathbb{V}}$.*

Důkaz. 1. $\Rightarrow$ 2. Tato implikace je triviální. Protože $\mathbb{V}$ a $\mathbb{W}$ jsou zig-zag isomorfní, existují zig-zag morfismy $f : \mathbb{V} \nrightarrow \mathbb{W}$ a $g : \mathbb{W} \nrightarrow \mathbb{V}$ takové, že $f \circ g = 1_{\mathbb{W}}$ a $g \circ f = 1_{\mathbb{V}}$. Obě zobrazení jsou však silnými homomorfismy podle lemmatu 3.3.9.

2. $\Rightarrow$ 1. Podle lemmatu 3.3.9 stačí ukázat, že obě zobrazení splňují podmínku (B). Ukážeme to pouze pro zobrazení $f$, pro zobrazení $g$ lze postupovat analogicky.

Vezměme akci $f(v) \xrightarrow{R} w'$ ve $\mathbb{W}$. Protože $g$ je silný homomorfismus a platí $g \circ f = 1_{\mathbb{V}}$, máme ve $\mathbb{V}$ akci $v \xrightarrow{R} g(w')$. Jediné, co zbývá ukázat, je rovnost $f(g(w')) = w'$. To ale plyne z faktu $f \circ g = 1_{\mathbb{W}}$. ■

Zig-zag isomorfní akční struktury považujeme za stejné, podobně jako považujeme za stejné isomorfní grafy. Zdůrazněme, že lemma 3.3.13 říká, že isomorfismus akčních struktur prostřednictvím silných homomorfismů vyjde nastejno, jako isomorfismus zprostředkovaný zig-zag morfismy. U dalšího důležitého pojmu *podstruktury*, však situace bude jiná.

Uvažujme o akčních strukturách $\mathbb{V}$ a $\mathbb{W}$ takových, že množina vrcholů $V$ je podmnožinou množiny vrcholů $W$. Jaké požadavky musí akční struktura $\mathbb{W}$ splňovat, aby zobrazení $f : W \to V$ takové, že $f(w) = w$, bylo zig-zag morfismem? (Srovnejte s lemmatem 2.1.5.)

Odpověď dává lemma 3.3.9. Zobrazení $f$ musí být:

1. Silný homomorfismus. To znamená, že je-li ve $\mathbb{V}$ akce $v \xrightarrow{R} v'$, pak je stejná akce i ve $\mathbb{W}$. Strukturu $\mathbb{V}$ si tedy lze představit jako výběr nějakých vrcholů z množiny $W$, ovšem mezi tyto vybrané vrcholy jsme nesměli žádnou akci, která by nebyla ve struktuře $\mathbb{W}$, přidat.

2. Zobrazení splňující podmínku (B). Tato podmínka v našem konkrétním případě znamená, že je-li ve $\mathbb{W}$ akce $v \xrightarrow{R} w'$ z vrcholu $v \in V$, pak vrchol $w'$ musí být také v množině $V$. Struktura $\mathbb{V}$ je tedy ve $\mathbb{W}$ „uzavřená na akce".

Předchozí úvaha nás vede k definici akční podstruktury.

**3.3.14 Definice** Řekneme, že akční struktura $\mathbb{V}$ je *akční podstrukturou akční struktury* $\mathbb{W}$, pokud zobrazení $f : V \to W$ takové, že $f(v) = v$ pro všechna $v \in V$, je zig-zag morfismus.

**3.3.15 Lemma** *Následující je pro zig-zag morfismus* $f : \mathbb{V} \twoheadrightarrow\!\!\!\!\!\rightarrow \mathbb{W}$ *ekvivalentní:*

1. *Zobrazení* $f$ *je prosté.*

2. *Existuje akční struktura* $\mathbb{W}'$ *taková, že* $\mathbb{W}'$ *je zig-zag isomorfní struktuře* $\mathbb{V}$ *a navíc* $\mathbb{W}'$ *je akční podstrukturou struktury* $\mathbb{W}$.

Důkaz. 1. $\Rightarrow$ 2. Strukturu $\mathbb{W}'$ definujeme následovně: množina vrcholů $W'$ je $\{f(v) \in W \mid v \in V\}$ a ve $\mathbb{W}'$ je akce $f(v) \xrightarrow{R} f(v')$ právě tehdy, když ve $\mathbb{V}$ je akce $v \xrightarrow{R} v'$. Zobrazení $f : \mathbb{V} \to \mathbb{W}'$ je zřejmě silný isomorfismus, protože zobrazení $f$ je prosté. Podle lemmatu 3.3.13 je tedy $f : \mathbb{V} \to \mathbb{W}'$ silný isomorfismus.

Stačí tedy ukázat, že $\mathbb{W}'$ je akční podstruktura $\mathbb{W}$. Definujme zobrazení $h : W' \to W$ předpisem $h(w) = w$. Podle definice 3.3.14 stačí ukázat, že $h$ je zig-zag morfismus. To ale plyne okamžitě z toho, že $f$ je zig-zag morfismus.

2. $\Rightarrow$ 1. Tato implikace je triviální. ■

Protože zig-zag isomorfní akční struktury lze považovat za totožné, jsou prosté zig-zag morfismy vpodstatě totéž jako podstruktury. Rozšíříme tedy definici 3.3.14 na pojem podobjektu. Navíc zavedeme k podobjektu i duální pojem — faktorový objekt.

**3.3.16 Definice** Ať $\mathbb{V}$ je akční struktura. Řekneme, že akční struktura $\mathbb{W}$ je

1. *podobjektem* struktury $\mathbb{V}$, pokud existuje prostý zig-zag morfismus $f : \mathbb{W} \twoheadrightarrow\!\!\!\!\!\rightarrow \mathbb{V}$.

2. *faktorovým objektem* struktury $\mathbb{V}$, pokud existuje surjektivní zig-zag morfismus $f : \mathbb{V} \twoheadrightarrow\!\!\!\!\!\rightarrow \mathbb{W}$.

## 3.4 Cvičení

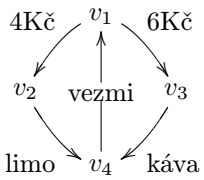
**3.4.1 Cvičení** Dokažte lemma 3.3.10.

**3.4.2 Cvičení** Dejte přesnou definici derivačního stromu akce v jazyce CCS.

**3.4.3 Cvičení** V jazyce CCS je popsána následující akční struktura $\mathbb{V}$ typu $\{4\text{Kč}, 6\text{Kč}, \text{káva}, \text{limo}, \text{vezmi}\}$:

$$\begin{aligned} v_1 &\stackrel{\text{def}}{=} 4\text{Kč}.v_2 + 6\text{Kč}.v_3 \\ v_2 &\stackrel{\text{def}}{=} \text{limo}.v_4 \\ v_3 &\stackrel{\text{def}}{=} \text{káva}.v_4 \\ v_4 &\stackrel{\text{def}}{=} \text{vezmi}.v_1 \end{aligned}$$

Ukažte, že obrázek akční struktury $\mathbb{V}$ je




a podejte derivační stromy všech akcí ve struktuře $\mathbb{V}$.

Strukturu $\mathbb{V}$ si lze představit jako jednoduchý automat na kávu a limonádu. Tento příklad pochází z [Mi89].

**3.4.4 Cvičení** V jazyce CCS je popsána následující akční struktura $\mathbb{V}$ typu $\{\text{zero}, \text{up}, \text{down}\}$ následovně:

$$\begin{aligned} c_0 &\stackrel{\text{def}}{=} \text{zero}.c_0 \\ c_{i+1} &\stackrel{\text{def}}{=} \text{up}.c_{i+2} + \text{down}.c_i, \quad \text{pro } i \geq 0 \end{aligned}$$

Namalujte obrázek akční struktury $\mathbb{V}$ a podejte derivační stromy všech akcí ve struktuře $\mathbb{V}$.

Strukturu $\mathbb{V}$ si lze představit jako jednoduchý čítač.

**3.4.5 Cvičení** Dejte příklad akčních struktur $\mathbb{V}$ a $\mathbb{W}$ takových, že platí:

1. Množina vrcholů $V$ akční struktury $\mathbb{V}$ je podmnožinou množiny vrcholů $W$ akční struktury $\mathbb{W}$.
2. $\mathbb{V}$ není akční podstrukturou $\mathbb{W}$.
3. Zobrazení $f : V \to W$, definované předpisem $f(v) = v$ pro všechna $v \in V$, je silný homomorfismus.

**3.4.6 Cvičení** Ukažte, že nelze obecně definovat pojem součinu akčních struktur (stejného typu). Přesněji: dejte příklad systému akčních struktur $\mathbb{V}_i$, $i \in I$ takových, že neexistuje akční struktura $\mathbb{V}$ a zig-zag morfismy $p^i : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\times\; \mathbb{V}_i$, $i \in I$, tak, aby byla splněna následující podmínka:

> Je-li $\mathbb{W}$ libovolná akční struktura a je-li $f^i : \mathbb{W} \rightarrowtail\!\!\!\!\!\!\times\; \mathbb{V}_i$, $i \in I$, libovolný systém zig-zag morfismů, pak existuje právě jeden zig-zag morfismus $f : \mathbb{W} \rightarrowtail\!\!\!\!\!\!\times\; \mathbb{V}$ tak, že $p^i \circ f = f^i$ pro všechna $i \in I$.

Srovnejte se lemmatem 2.1.4.

**3.4.7 Cvičení** Ukažte, že výsledek cvičení 3.4.6 platí, pokud zig-zag morfismy všude zaměníte za silné homomorfismy. Dokažte, tedy následující:

Je-li $\mathbb{V}_i$, $i \in I$, libovolný systém akčních struktur, pak existuje akční struktura $\mathbb{V}$ a silné homomorfismy $p^i : \mathbb{V} \to \mathbb{V}_i$, $i \in I$, takové, že je splněna následující podmínka:

> Je-li $\mathbb{W}$ libovolná akční struktura a je-li $f^i : \mathbb{W} \to \mathbb{V}_i$, $i \in I$, libovolný systém silných homomorfismů, pak existuje právě jeden silný homomorfismus $f : \mathbb{W} \to \mathbb{V}$ tak, že $p^i \circ f = f^i$ pro všechna $i \in I$.

Srovnejte s lemmatem 2.1.4.

**3.4.8 Cvičení** Ukažte, že výsledek cvičení 3.4.6 platí, pokud u všech zig-zag morfismů obrátíte směr. Dokažte, tedy následující:

Je-li $\mathbb{V}_i$, $i \in I$, libovolný systém akčních struktur, pak existuje akční struktura $\mathbb{V}$ a zig-zag morfismy $p^i : \mathbb{V}_i \to \mathbb{V}$, $i \in I$, takové, že je splněna následující podmínka:

> Je-li $\mathbb{W}$ libovolná akční struktura a je-li $f^i : \mathbb{V}_i \to \mathbb{W}$, $i \in I$, libovolný systém silných homomorfismů, pak existuje právě jeden silný homomorfismus $f : \mathbb{V} \to \mathbb{W}$ tak, že $f \circ p^i = f^i$ pro všechna $i \in I$.

(Návod: inspirujte se koproduktem grafů — definice 2.1.6.) Takto definovanou akční strukturu $\mathbb{V}$ nazýváme *koprodukt akčních struktur* $\mathbb{V}_i$ a značíme ji $\coprod_{i \in I} \mathbb{V}_i$.

# Kapitola 4

# Modální logika

Jazyk modální logiky, který zavedeme v této kapitole, nám umožní pronášet výroky o akčních strukturách. Modální logika rozšiřuje výrokovou logiku o možnost dát výroku *modalitu*. V klasické modální logice jde o modality *je možné, že* a *je nutné, že* — viz odstavec 1.1. V Hennessyho-Milnerově modální logice je systém modalit dán typem $\Delta$ pro akční struktury. Pro různé typy tedy dostáváme různě bohaté modální logiky. Formule Hennessyho-Milnerovy modální logiky jsou (pro daný pevný typ $\Delta$) vhodné pro vyjadřování se o akčních strukturách (typu $\Delta$). Akční struktury se navíc stanou přirozenou strukturou, ve které lze formule Hennessyho-Milnerovy modální logiky interpretovat. Interpretace formulí, kterou v této kapitole zavedeme, je zobecněním sémantiky možných světů pro formule klasické modální výrokové logiky (viz 1.1).

## 4.1 Hennessyho-Milnerova modální logika

Předpokládejme, že je dána množina $At$ (může být i prázdná). Množině $At$ říkáme *množina atomických formulí*. Dále předpokládejme, že je dán typ $\Delta$. Syntaxe formulí Hennessyho-Milnerovy modální logiky je dána následovně:

$$\varphi ::= a \mid \mathtt{tt} \mid (\neg\varphi) \mid (\varphi \wedge \varphi) \mid ([R]\varphi) \tag{4.1}$$

kde $a \in At$, $R \in \Delta$.

Výše uvedený zápis je takzvaná *Backusova-Naurova forma zápisu* induktivně definovaných objektů, často používaná v programování. Zápis (4.1) je třeba číst takto:

1. Zvolme $\varphi$ jako generické jméno pro formuli, $a$ jako generické jméno pro atomickou formuli a $R$ jako generické jméno pro akci.
2. Formule je
   (a) buď atomická formule $a \in At$,
   (b) nebo formule $\mathtt{tt}$,
   (c) nebo řetězec $(\neg\varphi)$, kde $\varphi$ je generické jméno formule,
   (d) nebo řetězec $(\varphi \wedge \varphi)$, kde $\varphi$ je generické jméno formule (Pozor! V tomto případě může být $\varphi$ napravo odlišné od $\varphi$ nalevo. Zanedlouho tuto notační nepřehlednost odstraníme.),
   (e) nebo řetězec $([R]\varphi)$, kde $R \in \Delta$ a $\varphi$ je generické jméno formule.
3. Formule vzniká pouze konečným použitím výše uvedených pravidel.

Množinu všech řetězců vytvořených podle výše uvedených pravidel označíme $Log_\Delta(At)$.

V praxi se často používá *relaxovaná BNF*, kde je dovoleno generická jména indexovat. Zápis je pak přehlednější:

$$\varphi ::= a \mid \mathtt{tt} \mid (\neg\varphi_1) \mid (\varphi_1 \wedge \varphi_2) \mid ([R]\varphi_1) \tag{4.2}$$

kde $a \in At$, $R \in \Delta$.

Formuli $\mathtt{tt}$ čteme *true*. Symboly $\neg$ a $\wedge$ čteme stejně jako ve výrokové logice, tj. *negace* a *konjunkce*. Řetězec $([R]\varphi)$ čteme *box* $R$ $\varphi$. Další spojky zavedeme jako *syntaktické zkratky*:

1. *False*: `ff` je zkratka za $(\neg\mathtt{tt})$.
2. *Disjunkce*: $(\varphi_1 \vee \varphi_2)$ je zkratka za $(\neg((\neg\varphi_1) \wedge (\neg\varphi_2)))$.
3. *Implikace*: $(\varphi_1 \Rightarrow \varphi_2)$ je zkratka za $((\neg\varphi_1) \vee \varphi_2)$.
4. *Ekvivalence*: $(\varphi_1 \Leftrightarrow \varphi_2)$ je zkratka za $((\varphi_1 \Rightarrow \varphi_2) \wedge (\varphi_2 \Rightarrow \varphi_1))$.
5. *Diamant*: $(\langle R\rangle\varphi)$ (čteme *diamant R* $\varphi$) je zkratka za $(\neg([R](\neg\varphi)))$.

Podobně jako ve výrokové logice učiníme konvenci o závorkách: *nepíšeme* nejvíce vnější závorky a předpokládáme, že negace, box a diamant *váží silněji*, než ostatní spojky. Tj. například namísto $(\neg([R](\neg\varphi)))$ píšeme $\neg[R]\neg\varphi$.

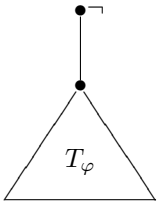
Analogicky jako ve výrokové logice, definujeme i zde pojem *syntaktického stromu* (také *derivačního stromu*):

1. Syntaktický strom atomické formule $a$ je:



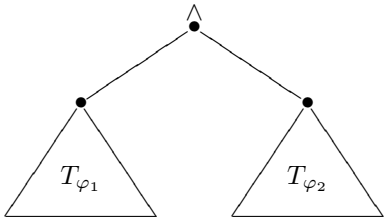

2. Syntaktický strom formule `tt` je:

3. Je-li $T_\varphi$ syntaktický strom formule $\varphi$, pak strom



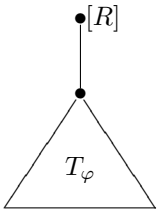

je syntaktický strom formule $\neg\varphi$.

4. Jsou-li $T_{\varphi_1}$ a $T_{\varphi_2}$ syntaktické stromy formulí $\varphi_1$ a $\varphi_2$, pak strom




je syntaktický strom formule $\varphi_1 \wedge \varphi_2$.

5. Je-li $T_\varphi$ syntaktický strom formule $\varphi$, pak strom




je syntaktický strom formule $[R]\varphi$.

**4.1.1 Definice** *Syntaktická složitost formule* $\varphi$ je výška syntaktického stromu formule $\varphi$.

## 4.2 Sémantika modálních formulí

Formule modální logiky $Log_\Delta(At)$ budeme interpretovat v akčních strukturách typu $\Delta$. Podobně jako ve výrokové logice musíme nejprve říci, jaké je ohodnocení atomických formulí.

Ve výrokové logice je modelem ohodnocení atomických formulí: nejprve se rozhodneme, které atomické formule jsou pravdivé a které nepravdivé. Pravidla pro ohodnocování výrokové logiky pak dají ohodnocení všech formulí výrokové logiky. (Viz také cvičení 4.4.7.)

Modelem formulí modální logiky bude akční struktura $\mathbb{V}$ typu $\Delta$ a vyznačení těch vrcholů ve $\mathbb{V}$, kde je daná atomická formule pravdivá. Zformulujme to přesně:

**4.2.1 Definice** *Modální model $\mathscr{M}$ logiky $Log_\Delta(At)$* je dvojice $\langle \mathbb{V}, \mathscr{V}\rangle$, kde $\mathbb{V}$ je akční struktura typu $\Delta$ a $\mathscr{V}$ je funkce z množiny $At$ do množiny $exp(V)$. Funkci $\mathscr{V}$ říkáme *valuace*.

Funkce $\mathscr{V}$ tedy každé atomické formuli $a$ přiřadí nějakou množinu vrcholů akční struktury $\mathbb{V}$. Intuice je zde taková, že v modelu $\mathscr{M}$ je atomická formule $a$ pravdivá přesně ve všech vrcholech z množiny $\mathscr{V}(a)$.

Tato situace bude typická: pojem pravdivosti modální formule budeme definovat lokálně, tj. vzhledem k jednomu vrcholu. Protože budeme chtít zavést ještě globální pojem pravdivosti, budeme z důvodu větší přehlednosti lokálnímu pojmu pravdivosti říkat *vynucení*.

**4.2.2 Definice** Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ je modální model logiky $Log_\Delta(At)$. Vztah $v \Vdash_{\mathscr{M}} \varphi$ *$\mathscr{M}$-vynucení formule $\varphi$ ve vrcholu $v$* je definován induktivně následovně:

1. $v \Vdash_{\mathscr{M}} a$ platí právě tehdy, když $v \in \mathscr{V}(a)$.
2. $v \Vdash_{\mathscr{M}} \mathtt{tt}$ platí vždy.
3. $v \Vdash_{\mathscr{M}} \neg\varphi$ platí právě tehdy, když neplatí $v \Vdash_{\mathscr{M}} \varphi$.
4. $v \Vdash_{\mathscr{M}} \varphi_1 \wedge \varphi_2$ platí právě tehdy, když platí $v \Vdash_{\mathscr{M}} \varphi_1$ a současně $v \Vdash_{\mathscr{M}} \varphi_2$.
5. $v \Vdash_{\mathscr{M}} [R]\varphi$ platí právě tehdy, když pro každý vrchol $v'$ platí: je-li $v \xrightarrow{R} v'$, pak $v' \Vdash_{\mathscr{M}} \varphi$.

Jestliže $v \Vdash_{\mathscr{M}} \varphi$ řekneme, že formule $\varphi$ je *$\mathscr{M}$-vynucena ve vrcholu $v$*. Fakt, že formule $\varphi$ *není $\mathscr{M}$-vynucena ve vrcholu $\varphi$*, značíme $v \nVdash_{\mathscr{M}} \varphi$.

**4.2.3 Lemma** *Pro odvozené spojky je $\mathscr{M}$-vynucení popsáno následovně:*

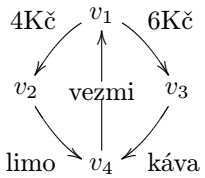
1. $v \Vdash_{\mathscr{M}} \mathtt{ff}$ *neplatí nikdy.*
2. $v \Vdash_{\mathscr{M}} \varphi_1 \vee \varphi_2$ *platí právě tehdy, když platí* $v \Vdash_{\mathscr{M}} \varphi_1$ *nebo* $v \Vdash_{\mathscr{M}} \varphi_2$.
3. $v \Vdash_{\mathscr{M}} \varphi_1 \Rightarrow \varphi_2$ *platí právě tehdy, když platí: je-li* $v \Vdash_{\mathscr{M}} \varphi_1$, *pak* $v \Vdash_{\mathscr{M}} \varphi_2$.
4. $v \Vdash_{\mathscr{M}} \langle R\rangle\varphi$ *platí právě tehdy, když existuje vrchol $v'$ a akce $v \xrightarrow{R} v'$ tak, že $v' \Vdash_{\mathscr{M}} \varphi$.*

Důkaz. Viz cvičení 4.4.4. ■

Globální pojem pravdivosti je definován následovně:

**4.2.4 Definice** Řekneme, že *modální formule $\varphi$ je pravdivá v modelu $\mathscr{M}$* a tento fakt značíme $\models_{\mathscr{M}} \varphi$, jestliže platí $v \Vdash_{\mathscr{M}} \varphi$ pro každý vrchol $v$. Fakt, že formule $\varphi$ *není pravdivá v modelu $\mathscr{M}$*, značíme $\not\models_{\mathscr{M}} \varphi$.

**4.2.5 Příklad** Připomeňme akční strukturu $\mathbb{V}$ ze cvičení 3.4.3:



Formule modální logiky můžeme použít k vyjádření některých vlastností akční struktury $\mathbb{V}$.

Modální model $\mathscr{M}$ na $\mathbb{V}$ definujme následovně: $\mathscr{V}(a) = \emptyset$ pro všechna $a \in At$. Řekli jsme tedy, že ve struktuře $\mathbb{V}$ není v žádném vrcholu $\mathscr{M}$-vynucena žádná atomická formule.

Příklady formulí, které jsou $\mathscr{M}$-vynuceny, jsou:

1. $v_1 \Vdash_{\mathscr{M}} \langle 4\text{Kč}\rangle \mathtt{tt}$. To lze intuitivně chápat tak, že ve stavu $v_1$ lze *vhodit čtyři koruny.*
2. $v_1 \Vdash_{\mathscr{M}} \langle 6\text{Kč}\rangle([\text{limo}]\mathtt{ff} \wedge \langle \text{káva}\rangle \mathtt{tt})$. To lze intuitivně chápat tak, že *po vhození šesti korun ve stavu $v_1$ lze chtít kávu, ale nikoli limonádu.*

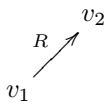
Viz také cvičení 4.4.5.

Upozorněme, že zatímco pojem $\mathscr{M}$-vynucení se chová tak, jak očekáváme, při práci s pojmem pravdivosti v modelu musíme být opatrnější, jak ukazuje i následující příklad.

**4.2.6 Příklad** Uvedeme příklad modálního modelu $\mathscr{M}$ a formulí $\varphi$ a $\psi$ tak, že jsou splněny následující dvě podmínky:

1. Jestliže $\models_{\mathscr{M}} \varphi$, pak $\models_{\mathscr{M}} \psi$.
2. $\not\models_{\mathscr{M}} \varphi \Rightarrow \psi$.

Budeme předpokládat, že $\Delta = \{R\}$, $At = \{a\}$. Akční struktura $\mathbb{V}$ typu $\Delta$ bude dána obrázkem:




Modální model $\mathscr{M}$ na $\mathbb{V}$ zadáme, pokud zadáme valuaci $\mathscr{V}$. Máme k dispozici pouze jedinou atomickou formuli $a$, definujme $\mathscr{V}(a) = \{v_1\}$.

Za formuli $\varphi$ zvolme $a$, za formuli $\psi$ zvolme formuli $[R]a$. Implikace

jestliže $\models_{\mathscr{M}} \varphi$, pak $\models_{\mathscr{M}} \psi$

platí z triviálních důvodů: neplatí totiž její předpoklad $\models_{\mathscr{M}} \varphi$. Podle naší volby modelu totiž formule $a$ není $\mathscr{M}$-vynucena ve vrcholu $v_2$.

Ukážeme však, že neplatí $\models_{\mathscr{M}} \varphi \Rightarrow \psi$. K tomu stačí najít alespoň jeden vrchol $v$ takový, že ve $v$ není $\mathscr{M}$-vynucena implikace $\varphi \Rightarrow \psi$.

Tímto vrcholem je zřejmě vrchol $v_1$. Platí $v_1 \Vdash_{\mathscr{M}} a$, ale neplatí $v_1 \Vdash_{\mathscr{M}} [R]a$.

## 4.3 Morfismy modelů modální logiky

Podobně jako pro akční struktury nás bude i pro modely zajímat pojem „správného zobrazení“ mezi dvěma modely. I u modelů začneme nejprve pojmem relace mezi dvěma modely.

**4.3.1 Definice** Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ a $\mathcal{N} = \langle \mathbb{W}, \mathscr{W}\rangle$ jsou dva modální modely. Zig-zag relace $C : \mathbb{V} \rightarrowtail\!\!\!\!\!\circ\; \mathbb{W}$ je *zig-zag relace z modelu $\mathscr{M}$ do modelu $\mathcal{N}$* a tento fakt značíme $C : \mathscr{M} \rightarrowtail\!\!\!\!\!\circ\; \mathcal{N}$, pokud pro každé vrcholy takové, že $v\ C\ w$ platí:

$v \Vdash_{\mathscr{M}} a$ právě tehdy, když $w \Vdash_{\mathcal{N}} a$ pro každé $a \in At$.

Je-li zig-zag relace $C : \mathscr{M} \rightarrowtail\!\!\!\!\!\circ\; \mathcal{N}$ funkce, mluvíme o *zig-zag morfismu z modelu $\mathscr{M}$ do modelu $\mathcal{N}$* a značíme jej $C : \mathscr{M} \rightarrowtail \mathcal{N}$.

Jsou-li dva vrcholy v modálních modelech v relaci zig-zag, vynucují podle definice zig-zag relace mezi dvěma modálními modely stejné atomické formule. Ukážeme, že si pak vynucují stejné libovolné formule.

**4.3.2 Tvrzení** *Předpokládejme, že $C : \mathscr{M} \rightarrowtail\!\!\!\!\!\circ\; \mathcal{N}$. Ať $v$ a $w$ jsou vrcholy takové, že $v\ C\ w$, pak pro libovolnou formuli $\varphi$ platí:*

*$v \Vdash_{\mathscr{M}} \varphi$ právě tehdy, když $w \Vdash_{\mathcal{N}} \varphi$.*

Důkaz. Indukcí podle syntaktické složitosti formule $\varphi$.

- $\varphi = a$. Potom podle definice 4.3.1 $v \Vdash_{\mathscr{M}} a$ právě tehdy, když $w \Vdash_{\mathcal{N}} a$.
- $\varphi = \mathtt{tt}$. Tento případ je triviální.
- $\varphi = \neg\varphi_1$. Předpokládejme, že pro formuli $\varphi_1$ tvrzení platí. Potom $v \Vdash_{\mathscr{M}} \neg\varphi_1$ právě tehdy, když neplatí $v \Vdash_{\mathscr{M}} \varphi_1$, což je právě tehdy, když neplatí $w \Vdash_{\mathcal{N}} \varphi_1$ a to je právě tehdy, když platí $w \Vdash_{\mathcal{N}} \neg\varphi_1$.
- $\varphi = \varphi_1 \wedge \varphi_2$. Předpokládejme, že pro formule $\varphi_1$ a $\varphi_2$ tvrzení platí. Potom $v \Vdash_{\mathscr{M}} \varphi_1 \wedge \varphi_2$ právě tehdy, když platí $v \Vdash_{\mathscr{M}} \varphi_1$ a $v \Vdash_{\mathscr{M}} \varphi_2$ současně. Podle indukčního předpokladu to je právě tehdy, když platí $w \Vdash_{\mathcal{N}} \varphi_1$ a $w \Vdash_{\mathcal{N}} \varphi_2$ současně, neboli, právě tehdy, když $w \Vdash_{\mathcal{N}} \varphi_1 \wedge \varphi_2$.
- $\varphi = [R]\varphi_1$. Předpokládejme, že pro formuli $\varphi_1$ tvrzení platí.

  Ukážeme, že když platí $v \Vdash_{\mathscr{M}} [R]\varphi_1$, pak platí $w \Vdash_{\mathcal{N}} [R]\varphi_1$. Ať $w \overset{R}{\to} w'$ je libovolná akce. Podle podmínky (B) pro zig-zag relace existuje $v'$ takové, že $v \overset{R}{\to} v'$ a navíc $v' \; C \; w'$. Podle indukčního předpokladu aplikovaného na formuli $\varphi_1$ a vrcholy $v'$, $w'$ je pak $w' \Vdash_{\mathcal{N}} \varphi_1$. Použili jsme totiž faktu, že $v' \Vdash_{\mathscr{M}} \varphi_1$.

  Obrácená implikace se ukáže analogicky.

■

Podobně jako pro akční struktury i pro modální modely zavedeme pojem isomorfních modelů, podmodelu a faktorového modelu.

**4.3.3 Definice** Ať $\mathscr{M}$ a $\mathcal{N}$ jsou dva modální modely. Řekneme, že:

1. $\mathscr{M}$ a $\mathcal{N}$ jsou *isomorfní*, pokud existují zig-zag morfismy $f : \mathscr{M} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathcal{N}$ a $g : \mathcal{N} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathscr{M}$ takové, že $g \circ f = 1_{\mathscr{M}}$ a $f \circ g = 1_{\mathcal{N}}$.
2. $\mathscr{M}$ je *podmodelem* $\mathcal{N}$, pokud existuje prostý zig-zag morfismus $f : \mathscr{M} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathcal{N}$.
3. $\mathcal{N}$ je *faktorovým modelem* $\mathscr{M}$, pokud existuje surjektivní zig-zag morfismus $f : \mathscr{M} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathcal{N}$.

Následující lemma charakterisuje isomorfní modely.

**4.3.4 Lemma** *Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$ a $\mathcal{N} = \langle \mathbb{W}, \mathscr{W} \rangle$ jsou dva modální modely. Pro zig-zag morfismus $f : \mathscr{M} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathcal{N}$ je následující ekvivalentní:*

1. *$f$ je isomorfismus modelů.*
2. *$f : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathbb{W}$ je zig-zag isomorfismus a pro valuace $\mathscr{V}$ a $\mathscr{W}$ platí: $\mathscr{V}(a) = \{v \mid f(v) \in \mathscr{W}(a)\}$.*

Důkaz. 1. $\Rightarrow$ 2. Zobrazení $f$ je zřejmě zig-zag isomorfismus odpovídajících akčních struktur. Podle definice 4.3.1 pro každé $a \in At$ platí $v \Vdash_{\mathscr{M}} a$ právě tehdy, když platí $f(v) \Vdash_{\mathcal{N}} a$.

2. $\Rightarrow$ 1. Podle předpokladu existuje zig-zag morfismus $g : \mathbb{W} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathbb{V}$ takový, že $g \circ f = 1_{\mathbb{V}}$ a $f \circ g = 1_{\mathbb{W}}$. Protože víme, že $f$ je zig-zag morfismus modelů, stačí tedy ukázat, že $g$ je zig-zag morfismus modelů. K tomu je potřeba ukázat, že pro libovolný vrchol $w$ a pro každé $a \in At$ platí $w \Vdash_{\mathcal{N}} a$ právě tehdy, když platí $g(w) \Vdash_{\mathscr{M}} a$.

Označme $v = g(w)$. Tvrzení pak plyne z toho, že $f$ je zig-zag morfismus modelů a z toho, že platí $f(g(w)) = w$ a $g(f(v)) = v$. ■

Jestliže máme modální model $\langle \mathbb{W}, \mathcal{W} \rangle$ a akční strukturu $\mathbb{V}$, pak libovolný zig-zag morfismus $f : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\rightarrow \mathbb{W}$ vytváří modální model na akční struktuře $\mathbb{V}$. Popis tohoto modelu uvádíme v následující definici.

**4.3.5 Definice** Ať $\mathbb{V}$ a $\mathbb{W}$ jsou dvě akční struktury stejného typu. Ať $f$ je jakékoli zobrazení z množiny vrcholů akční struktury $\mathbb{V}$ do množiny vrcholů akční struktury $\mathbb{W}$. Mějme dále libovolný modální model $\mathscr{M} = \langle \mathbb{W}, \mathscr{W} \rangle$.

Model $\langle \mathbb{V}, \mathscr{V} \rangle$, kde pro všechna $a \in At$ je

$$\mathscr{V}(a) = \{v \mid f(v) \in \mathscr{W}(a)\}$$

nazveme *pullback modelu* $\mathscr{M}$ *podél* $f$ a označíme jej $f^{-1}(\mathscr{M})$.

**4.3.6 Tvrzení** *Ať* $f : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\leftrightarrow \mathbb{W}$ *je zig-zag morfismus akčních struktur a ať* $\mathscr{M} = \langle \mathbb{W}, \mathscr{W} \rangle$ *je libovolný modální model na* $\mathbb{W}$. *Potom* $f : f^{-1}(\mathscr{M}) \rightarrowtail\!\!\!\!\!\!\leftrightarrow \mathscr{M}$.

Důkaz. Stačí ukázat, že pro každý vrchol $v$ a každou atomickou formuli $a$ platí:

$v \Vdash_{f^{-1}(\mathscr{M})} a$ právě tehdy, když $f(v) \Vdash_{\mathscr{M}} a$.

$v \Vdash_{f^{-1}(\mathscr{M})} a$ platí právě tehdy, když $f(v) \in \mathscr{W}(a)$ podle definice pullbacku modelu $\mathscr{M}$. To ovšem znamená právě tolik, že platí $f(v) \Vdash_{\mathscr{M}} a$. ■

Kombinací předchozího tvrzení a tvrzení 4.3.2 dostáváme:

**4.3.7 Důsledek** *Ať* $f : \mathbb{V} \rightarrowtail\!\!\!\!\!\!\leftrightarrow \mathbb{W}$ *je zig-zag morfismus akčních struktur a ať* $\mathscr{M} = \langle \mathbb{W}, \mathscr{W} \rangle$ *je libovolný modální model na* $\mathbb{W}$. *Potom pro libovolný vrchol* $v$ *ve* $\mathbb{V}$ *a pro libovolnou formuli* $\varphi$ *platí:*

$v \Vdash_{f^{-1}(\mathscr{M})} \varphi$ *právě tehdy, když* $f(v) \Vdash_{\mathscr{M}} \varphi$.

Definujeme nyní duální pojem k pullbacku modelu. Připomeňme, že pro zobrazení $f : V \to W$ a prvek $w \in W$ značí $f^{-1}(w)$ množinu $\{v \in V \mid f(v) = w\}$.

**4.3.8 Definice** Ať $f$ je jakékoli zobrazení z množiny vrcholů akční struktury $\mathbb{V}$ do množiny vrcholů akční struktury $\mathbb{W}$. Mějme dále libovolný modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$.

Model $\langle \mathbb{W}, \mathscr{W} \rangle$, kde pro všechna $a \in At$ je

$$\mathscr{W}(a) = \{w \mid f^{-1}(w) \subseteq \mathscr{V}(a)\}$$

nazveme *rozšířením modelu* $\mathscr{M}$ *podél* $f$ a označíme jej $f(\mathscr{M})$.

**4.3.9 Lemma** *Předpokládejme, že* $f : \mathbb{W} \rightarrowtail\!\!\!\!\!\!\leftrightarrow \mathbb{V}$ *je prostý zig-zag morfismus. Ať* $\mathscr{M} = \langle \mathbb{W}, \mathscr{W} \rangle$ *je modální model. Model* $f(\mathscr{M})$ *označme jako* $\mathscr{N} = \langle \mathbb{V}, \mathscr{V} \rangle$. *Potom platí* $f^{-1}(\mathscr{N}) = \mathscr{M}$.

Důkaz. Označme $f^{-1}(\mathscr{N}) = \langle \mathbb{W}, \mathscr{W}' \rangle$. Stačí ukázat, že pro libovolné $a \in At$ platí $\mathscr{W}(a) = \mathscr{W}'(a)$. To ale plyne z toho, že $f$ je prosté zobrazení: pro každý vrchol $v$ totiž platí $f^{-1}(f(v)) = \{v\}$. ■

## 4.4 Cvičení

**4.4.1 Cvičení** Popište algoritmus, který pro řetězec znaků sestávající z prvků množiny $At$, závorek logických spojek, boxů a diamantů, rozhodne, zda se jedná o formuli. (Návod: využijte pojmu syntaktického stromu.)

**4.4.2 Cvičení** Popište algoritmus, který formuli Hennessyho-Milnerovy modální logiky přepíše do:

prefixního zápisu,

postfixního zápisu.

(Návod: využijte pojmu syntaktického stromu.)

**4.4.3 Cvičení** Předpokládejte, že modální formule $\varphi$ neobsahuje žádnou atomickou formuli. Ukažte, že pak lze $\varphi$ napsat v jednom z následujících tvarů:

1. `tt`.
2. `ff`.
3. $\varphi_1 \wedge \varphi_2$.
4. $\varphi_1 \vee \varphi_2$.
5. $[R]\varphi_1$.
6. $\langle R\rangle\varphi_1$.

**4.4.4 Cvičení** Dokažte lemma 4.2.3.

**4.4.5 Cvičení** Nalezněte příklady dalších formulí, které jsou $\mathscr{M}$-vynuceny v modelu automatu na kávu a limonádu z příkladu 4.2.5. Dejte těmto formulím intuitivní interpretaci.

**4.4.6 Cvičení** Označte akční strukturu popisující automat na kávu a limonádu z příkladu 4.2.5 jako $\mathbb{V}$. Předpokládejte, že $a \in At$. Na $\mathbb{V}$ definujte valuaci $\mathscr{V}$ takto: $\mathscr{V}(a) = \{v_2, v_4\}$, $\mathscr{V}(b) = \emptyset$ pro $b \neq a$. Výsledný modální model označte $\mathscr{M}$ a ukažte, že platí:

1. $\models_{\mathscr{M}} [4\text{Kč}]a \wedge [\text{vezmi}]\neg a$.
2. $\models_{\mathscr{M}} a \Rightarrow [\text{limo}]a$.
3. $\not\models_{\mathscr{M}} a \vee \langle 4\text{Kč}\rangle a$.
4. $\not\models_{\mathscr{M}} a \Rightarrow \langle 4\text{Kč}\rangle a$.

**4.4.7 Cvičení** Modální výroková logika zobecňuje výrokovou logiku v následujícím smyslu: Ať $\Delta = \emptyset$. Potom platí:

1. Formule v $Log_\emptyset(At)$ jsou přesně formule klasické výrokové logiky nad množinou atomických formulí $At$. Dokažte.
2. Akční struktury pro $\Delta = \emptyset$ jsou přesně množiny. Dokažte.
3. Popište relaci vynucení a modální modely pro $\Delta$.
4. Ukažte, že pokud má akční struktura $\mathbb{V}$ pouze jeden vrchol, dejme tomu $v$, pak modální modely $\mathscr{M}$ na $\mathbb{V}$ přesně odpovídají ohodnocením ve výrokové logice. (Návod: Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ je modální model na $\mathbb{V}$. Definujte ohodnocení $u$ takto: $u(\varphi) = 1$ iff $v \Vdash_{\mathscr{M}} \varphi$. Analogicky postupujte při přechodu od ohodnocení k modálnímu modelu na $\mathbb{V}$.)

**4.4.8 Cvičení** Ukažte, že pro libovolný modální model $\mathscr{M}$ a libovolnou modální formuli $\varphi$ platí:

Jestliže platí $\models_{\mathscr{M}} \neg\varphi$, potom $\models_{\mathscr{M}} \varphi$ neplatí.

Obrácení výše uvedené implikace však neplatí: ukažte, že existuje modální model $\mathscr{M}$ a modální formule $\varphi$ tak, že neplatí ani $\models_{\mathscr{M}} \varphi$ ani $\models_{\mathscr{M}} \neg\varphi$.

**4.4.9 Cvičení** Připomeňme, že nad množinu akcí $\Delta$ lze vytvořit *volný monoid* $\Delta^*$. Prvky množiny $\Delta^*$ jsou konečná slova nad abecedou $\Delta$. Operace zřetězování slov je zřejmě asociativní a prázdné slovo $\varepsilon$ se chová jako neutrální prvek k této operaci.

Typický prvek množiny $\Delta^*$ budeme značit $\mathscr{R}$. Definujte *zobecněné modality* $[\mathscr{R}]\varphi$ a $\langle\mathscr{R}\rangle\varphi$ induktivně takto:

1. $[\varepsilon]\varphi = \varphi$.
2. $[\mathscr{R}]\varphi = [R]([\mathscr{R}']\varphi)$, je-li $\mathscr{R} = R\mathscr{R}'$.
3. $\langle\varepsilon\rangle\varphi = \varphi$.
4. $\langle\mathscr{R}\rangle\varphi = \langle R\rangle(\langle\mathscr{R}'\rangle\varphi)$, je-li $\mathscr{R} = R\mathscr{R}'$.

Odvoďte následující vztahy:

1. $v \Vdash_{\mathscr{M}} [\mathscr{R}]\varphi$ platí právě tehdy, když pro libovolnou akci $v \xrightarrow{\mathscr{R}} v'$ platí $v \Vdash_{\mathscr{M}} \varphi$.
2. $v \Vdash_{\mathscr{M}} \langle\mathscr{R}\rangle\varphi$ platí právě tehdy, když existuje akce $v \xrightarrow{\mathscr{R}} v'$ taková, že platí $v \Vdash_{\mathscr{M}} \varphi$.

# Kapitola 5

# Systémy modální logiky

V této kapitole se budeme zajímat o modální formule, které platí ve všech modelech modální logiky, nebo alespoň ve velké třídě modálních modelů. Takové formule nevypovídají příliš mnoho o akčních strukturách. Příkladem formule, která je pravdivá ve všech modelech modální logiky, je formule $\mathtt{tt}$. Jiným příkladem modální formule, která je pravdivá ve všech modálních modelech, je formule $\varphi \vee \neg\varphi$, kde $\varphi$ je libovolná modální formule. Takovým modálním formulím budeme říkat modální tautologie a takové formule jsou „nezajímavé“ — protože jsou pravdivé v každém modálním modelu, nemohou totiž nic o akčních strukturách vypovědět. V této kapitole podáme syntaktickou charakterisaci modálních tautologií.

Některé formule Hennessyho-Milnerovy modální logiky však nejsou modální tautologie, přesto jsou pravdivé v každém modálním modelu nad velkou třídou akčních struktur. Například formule $[R]\varphi \Rightarrow [R][R]\varphi$ (kde $\varphi$ je libovolná modální formule) je pravdivá přesně ve všech modálních modelech na akčních strukturách, kde akce $R$ je interpretována jako transitivní binární relace. O formuli $[R]\varphi \Rightarrow [R][R]\varphi$ pak můžeme říci, že *modálně definuje* transitivitu $R$. Věta o korespondenci (věta 5.2.3), kterou v této kapitole dokážeme, umožní alespoň pro některé jednoduché vlastnosti akčních struktur rozhodnout, zda jsou modálně definovatelné. Ukážeme však, že existují i takové vlastnosti akčních struktur, které modálně definovatelné nejsou.

## 5.1 Systém K — minimální modální logika

Připomeňme definici 4.2.4: modální formule $\varphi$ je pravdivá v modelu $\mathscr{M}$, jestliže platí $v \Vdash_{\mathscr{M}} \varphi$ pro každý vrchol $v$. Tento fakt značíme $\models_{\mathscr{M}} \varphi$.

**5.1.1 Definice** Řekneme, že formule $\varphi$ je *modální tautologie* a tento fakt značíme $\models \varphi$, pokud $\models_{\mathscr{M}} \varphi$ pro libovolný modální model $\mathscr{M}$.

V dalším budeme modální tautologie syntakticky charakterisovat. Lze ukázat, že modální tautologie jsou přesně ty modální formule, které lze odvodit z jisté sady axiomů předem danými odvozovacími pravidly. Tento deduktivní systém (tj. tuto sadu axiomů a odvozovacích pravidel) nazveme *systém* **K** nebo také *minimální modální logika*.

**5.1.2 Definice** Systém **K** obsahuje následující axiomy a odvozovací pravidla:

*Axiomy* jsou dvou typů:

1. Instance libovolné ze čtyř následujících formulí je axiom, kde $\varphi$, $\psi$, $\psi_1$ a $\psi_2$ jsou libovolné formule modální logiky:
   (a) $\mathtt{tt}$
   (b) $\varphi \Rightarrow (\psi \Rightarrow \varphi)$
   (c) $(\varphi \Rightarrow (\psi_1 \Rightarrow \psi_2)) \Rightarrow ((\varphi \Rightarrow \psi_1) \Rightarrow (\varphi \Rightarrow \psi_2))$
   (d) $(\neg\varphi \Rightarrow \neg\psi) \Rightarrow (\psi \Rightarrow \varphi)$
2. Instance následující formule je axiom, kde $\varphi$ je libovolná formule modální logiky:
   (a) $[R](\varphi \Rightarrow \psi) \Rightarrow ([R]\varphi \Rightarrow [R]\psi)$

*Odvozovací pravidla* jsou dvou typů, kde $\varphi$ a $\psi$ jsou libovolné formule modální logiky:

1. *Modus Ponens*:
   (a) Z formulí $\varphi \Rightarrow \psi$ a $\varphi$ odvoďte formuli $\psi$.
2. *Zavedení boxu*:
   (a) Z formule $\varphi$ odvoďte formuli $[R]\varphi$.

Fakt, že formule $\varphi$ je odvozena v deduktivním systému **K** budeme značit $\mathbf{K} \vdash \varphi$.

Formule, odvoditelné v systému **K**, jsou právě modální tautologie.

**5.1.3 Věta** *Pro libovolnou formuli $\varphi$ platí $\mathbf{K} \vdash \varphi$ právě tehdy, když $\models \varphi$.*

Důkaz. Důkaz rozdělíme na dvě části.

**I** Ukážeme, že každá formule odvozená v systému **K** je modální tautologie. Tomuto tvrzení se též říká *Věta o korektnosti*. Zřejmě stačí ukázat následující:

1. Každý axiom systému **K** je modální tautologie.
   (a) Zřejmě $\models \mathtt{tt}$.
   (b) Vezměme formuli $\varphi \Rightarrow (\psi \Rightarrow \varphi)$, libovolný model $\mathscr{M}$ a libovolný vrchol $v$.
   Předpokládejme, že platí $v \Vdash_{\mathscr{M}} \varphi$. Abychom ukázali, že platí $v \Vdash_{\mathscr{M}} \psi \Rightarrow \varphi$, předpokládejme, že platí $v \Vdash_{\mathscr{M}} \psi$. Pak ale platí $v \Vdash_{\mathscr{M}} \varphi$.
   (c) Vezměme formuli $(\varphi \Rightarrow (\psi_1 \Rightarrow \psi_2)) \Rightarrow ((\varphi \Rightarrow \psi_1) \Rightarrow (\varphi \Rightarrow \psi_2))$, libovolný model $\mathscr{M}$ a libovolný vrchol $v$.
   Předpokládejme, že platí
   $$v \Vdash_{\mathscr{M}} (\varphi \Rightarrow (\psi_1 \Rightarrow \psi_2)). \tag{5.1}$$
   Abychom ukázali, že platí
   $$v \Vdash_{\mathscr{M}} ((\varphi \Rightarrow \psi_1) \Rightarrow (\varphi \Rightarrow \psi_2)), \tag{5.2}$$
   předpokládejme, že platí
   $$v \Vdash_{\mathscr{M}} \varphi \Rightarrow \psi_1. \tag{5.3}$$
   Stačí ukázat, že jestliže platí
   $$v \Vdash_{\mathscr{M}} \varphi, \tag{5.4}$$
   pak platí
   $$v \Vdash_{\mathscr{M}} \psi_2. \tag{5.5}$$
   Podle (5.1) a (5.4) ale platí
   $$v \Vdash_{\mathscr{M}} \psi_1 \Rightarrow \psi_2 \tag{5.6}$$
   a podle (5.3) a (5.4) platí
   $$v \Vdash_{\mathscr{M}} \psi_1. \tag{5.7}$$
   Z platnosti (5.7) a (5.6) pak plyne platnost (5.5).
   (d) Vezměme formuli $(\neg\varphi \Rightarrow \neg\psi) \Rightarrow (\psi \Rightarrow \varphi)$, libovolný model $\mathscr{M}$ a libovolný vrchol $v$.
   Předpokládejme, že platí
   $$v \Vdash_{\mathscr{M}} \neg\varphi \Rightarrow \neg\psi. \tag{5.8}$$
   Ukážeme, že pak platí
   $$v \Vdash_{\mathscr{M}} \psi \Rightarrow \varphi. \tag{5.9}$$
   K tomu stačí předpokládat, že platí
   $$v \Vdash_{\mathscr{M}} \psi \tag{5.10}$$
   a ukázat, že platí
   $$v \Vdash_{\mathscr{M}} \varphi. \tag{5.11}$$

Budeme postupovat sporem, kdyby (5.11) neplatilo, musí platit

$$v \Vdash_{\mathscr{M}} \neg\varphi. \tag{5.12}$$

Z (5.12) a (5.8) plyne, že platí

$$v \Vdash_{\mathscr{M}} \neg\psi. \tag{5.13}$$

To je spor s (5.10).

(e) Vezměme formuli $[R](\varphi \Rightarrow \psi) \Rightarrow ([R]\varphi \Rightarrow [R]\psi)$, libovolný model $\mathscr{M}$ a libovolný vrchol $v$. Předpokládejme, že platí

$$v \Vdash_{\mathscr{M}} [R](\varphi \Rightarrow \psi). \tag{5.14}$$

Ukážeme, že pak platí

$$v \Vdash_{\mathscr{M}} [R]\varphi \Rightarrow [R]\psi. \tag{5.15}$$

K tomu stačí předpokládat, že platí

$$v \Vdash_{\mathscr{M}} [R]\varphi \tag{5.16}$$

a ukázat, že platí

$$v \Vdash_{\mathscr{M}} [R]\psi. \tag{5.17}$$

Vezměme libovolnou akci $v \xrightarrow{R} v'$. Potřebujeme ukázat, že

$$v' \Vdash_{\mathscr{M}} \psi. \tag{5.18}$$

Podle (5.14) platí

$$v' \Vdash_{\mathscr{M}} \varphi \Rightarrow \psi \tag{5.19}$$

a podle (5.16) platí

$$v' \Vdash_{\mathscr{M}} \varphi. \tag{5.20}$$

Ze (5.19) a (5.20) tedy plyne (5.18).

2. Každé z odvozovacích pravidel odvodí z modálních tautologií opět modální tautologie.

(a) Pravidlo Modus Ponens. Předpokládejme, že platí

$$\models \varphi \Rightarrow \psi \tag{5.21}$$

a

$$\models \varphi. \tag{5.22}$$

Vezměme libovolný model $\mathscr{M}$ a libovolný vrchol $v$. Podle (5.21) platí

$$v \Vdash_{\mathscr{M}} \varphi \Rightarrow \psi \tag{5.23}$$

a podle (5.22) platí

$$v \Vdash_{\mathscr{M}} \varphi. \tag{5.24}$$

Z (5.23) a (5.24) tedy dostáváme $v \Vdash_{\mathscr{M}} \psi$, což jsme měli dokázat.

(b) Pravidlo zavedení boxu. Předpokládejme, že platí

$$\models \varphi. \tag{5.25}$$

Vezměme libovolný model $\mathscr{M}$ a libovolný vrchol $v$. Abychom ukázali, že

$$v \Vdash_{\mathscr{M}} [R]\varphi, \tag{5.26}$$

vezměme libovolnou akci $v \xrightarrow{R} v'$. Podle (5.25) ale platí $v' \Vdash_{\mathscr{M}} \varphi$, což jsme měli ukázat.

**II** Důkaz toho, že každá modální tatutologie je odvoditelná v systému **K** odložíme do kapitoly 7, věta 7.6.5. Tomuto tvrzení se též říká *Věta o úplnosti*. ■

## 5.2 Věta o korespondenci

Modální tautologie jsou přesně ty formule, které jsou pravdivé ve všech modálních modelech. Zavedeme nyní pojem, který leží někde uprostřed pojmů „být pravdivý v jednom modelu“ a „být pravdivý ve všech modelech“.

**5.2.1 Definice** Ať $\Lambda$ je nějaká třída akčních struktur. Řekneme, že formule $\varphi$ je $\Lambda$-*pravdivá* a tento fakt značíme $\models_\Lambda \varphi$, pokud platí:

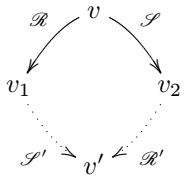
$\models_{\mathscr{M}} \varphi$ pro libovolný model $\mathscr{M}$ na libovolné akční struktuře ze třídy $\Lambda$.

Obsahuje-li $\Lambda$ pouze strukturu $\mathbb{V}$, budeme místo $\models_{\{\mathbb{V}\}} \varphi$ psát $\models_{\mathbb{V}} \varphi$.

Připomeňme zobeněné akce a modality typu $\Delta^*$ ze cvičení 4.4.9.

**5.2.2 Definice** Akční struktura $\mathbb{V}$ splňuje *zobecněnou Churchovu-Rosserovu vlastnost* $G(\mathscr{R},\mathscr{S}',\mathscr{S},\mathscr{R}')$, pokud pro libovolný vrchol $v$ platí:

Je-li $v \xrightarrow{\mathscr{R}} v_1$ a $v \xrightarrow{\mathscr{S}} v_2$, pak existuje vrchol $v'$ tak, že $v_1 \xrightarrow{\mathscr{S}'} v'$ a $v_2 \xrightarrow{\mathscr{R}'} v'$.




Třídu akčních struktur splňujících vlastnost $G(\mathscr{R},\mathscr{S}',\mathscr{S},\mathscr{R}')$ budeme značit $\Lambda G(\mathscr{R},\mathscr{S}',\mathscr{S},\mathscr{R}')$.

**5.2.3 Věta Věta o korespondenci.** *Předpokládejme, že $At \neq \emptyset$. Pro akční strukturu $\mathbb{V}$ je následující ekvivalentní:*

1. $\mathbb{V} \in \Lambda G(\mathscr{R},\mathscr{S}',\mathscr{S},\mathscr{R}')$.
2. $\models_{\mathbb{V}} \langle\mathscr{R}\rangle[\mathscr{S}']\varphi \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi$ *pro libovolnou modální formuli* $\varphi$.

Důkaz. 1. $\Rightarrow$ 2. Vezměme libovolnou formuli $\varphi$, libovolný modální model $\mathscr{M}$ na $\mathbb{V}$ a předpokládejme, že platí

$$v \Vdash_{\mathscr{M}} \langle\mathscr{R}\rangle[\mathscr{S}']\varphi. \tag{5.27}$$

Chceme ukázat, že potom platí

$$v \Vdash_{\mathscr{M}} [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi. \tag{5.28}$$

Zvolme libovolnou akci $v \xrightarrow{\mathscr{S}} v_2$. Protože předpokládáme (5.27) víme, že existuje akce $v \xrightarrow{\mathscr{R}} v_1$ taková, že platí

$$v_1 \Vdash_{\mathscr{M}} [\mathscr{S}']\varphi. \tag{5.29}$$

Protože struktura $\mathbb{V}$ splňuje podmínku $G(\mathscr{R},\mathscr{S}',\mathscr{S},\mathscr{R}')$, existuje vrchol $v'$ a akce $v_1 \xrightarrow{\mathscr{S}'} v'$ a $v_2 \xrightarrow{\mathscr{R}'} v'$. Podle (5.29) platí $v' \Vdash_{\mathscr{M}} \varphi$ a tudíž platí $v_2 \Vdash_{\mathscr{M}} \langle\mathscr{R}'\rangle\varphi$.

2. $\Rightarrow$ 1. Zvolme libovolný vrchol $v$ a libovolné akce $v \xrightarrow{\mathscr{R}} v_1$, $v \xrightarrow{\mathscr{S}} v_2$.
Na $\mathbb{V}$ definujeme následující model $\mathscr{M} = \langle\mathbb{V},\mathscr{V}\rangle$: zvolíme pevně jednu atomickou formuli $a$ a definujeme

$$\mathscr{V}(a) = \{w \mid \text{existuje akce } v_1 \xrightarrow{\mathscr{S}'} w\}, \tag{5.30}$$
$$\mathscr{V}(b) = \emptyset, \text{ pro } b \neq a. \tag{5.31}$$

Předpokládáme, že platí $\models_{\mathbb{V}} \langle\mathscr{R}\rangle[\mathscr{S}']\varphi \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi$ pro libovolnou formuli $\varphi$, tudíž speciálně platí $\models_{\mathscr{M}} \langle\mathscr{R}\rangle[\mathscr{S}']a \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle a$. Navíc díky naší volbě modelu platí $v \Vdash_{\mathscr{M}} \langle\mathscr{R}\rangle[\mathscr{S}']a$ a důsledkem toho je, že platí $v \Vdash_{\mathscr{M}} [\mathscr{S}]\langle\mathscr{R}'\rangle a$.

Protože $v \xrightarrow{\mathscr{S}} v_2$, platí $v_2 \Vdash_{\mathscr{M}} \langle\mathscr{R}'\rangle a$ a tudíž existuje akce $v_2 \xrightarrow{\mathscr{R}'} v'$ taková, že $v' \Vdash_{\mathscr{M}} a$, neboli množina $\mathscr{V}(a)$ je neprázdná, protože $v' \in \mathscr{V}(a)$.

Struktura $\mathbb{V}$ tedy splňuje zobecněnou Churchovu-Rosserovu vlastnost. ■

Povšimněme si, že v důkazu implikace 1. ⇒ 2. věty o korespondenci nebyl předpoklad $At \neq \emptyset$ použit. Platí tedy vždy následující tvrzení.

**5.2.4 Tvrzení** *Je-li* $\mathbb{V} \in \Lambda G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$, *pak* $\models_{\mathbb{V}} \langle\mathscr{R}\rangle[\mathscr{S}']\varphi \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi$ *pro libovolnou modální formuli* $\varphi$.

Pro protipříklad na implikaci 2. ⇒ 1. věty o korespondenci za situace $At = \emptyset$ odkazujeme na cvičení 5.4.4.

Formule platné ve všech strukturách splňujících zobecněnou Churchovu-Rosserovu vlastnost lze charakterisovat syntakticky. Přidejme formuli $\langle\mathscr{R}\rangle[\mathscr{S}']\varphi \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi$ jako další axiom k systému **K** a výsledný deduktivní systém označme $\mathbf{KG}(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$. Pak lze dokázat následující větu (viz cvičení 7.7.12).

**5.2.5 Věta** *Pro libovolnou formuli* $\varphi$ *platí* $\mathbf{KG}(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}') \vdash \varphi$ *právě tehdy, když* $\models_{\Lambda G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')} \varphi$.

## 5.3 Modální definovatelnost

**5.3.1 Definice** Ať $\Phi$ je nějaká vlastnost akčních struktur typu $\Delta$. Řekneme, že vlastnost $\Phi$ je *modálně definovatelná logikou* $Log_\Delta(At)$, pokud existuje formule $\varphi \in Log_\Delta(At)$ taková, že platí

$\mathbb{V}$ má vlastnost $\Phi$ právě tehdy, když $\models_{\mathbb{V}} \varphi$.

**5.3.2 Příklad** Věta o korespondenci 5.2.3 nám dává velkou třídu vlastností, které jsou modálně definovatelné v případě, že $At \neq \emptyset$. Jsou to například tyto vlastnosti:

1. Symetrie relace $R$. Odpovídá vlastnosti $G(R, R, \varepsilon, \varepsilon)$.
2. Reflexivita relace $R$. Odpovídá vlastnosti $G(\varepsilon, R, \varepsilon, \varepsilon)$.

Věta 5.2.3 tak vystihuje řadu důležitých vlastností akčních struktur, které lze modálně definovat. Uvedeme nyní zajímavý příklad modální definovatelnosti, který větou 5.2.3 vystihnout nelze.

**5.3.3 Příklad** Připomeňme, že binární relace $S$ je *reflexivní a transitivní uzávěr relace* $R$, pokud platí

$v \xrightarrow{S} v'$ platí právě tehdy, když platí $v \xrightarrow{R^n} v'$ pro nějaké $n \geq 0$,

kde $R^n$ značí provedení $n$ akcí $R$ za sebou (zápisem $R^0$ rozumíme akci $\varepsilon$) — viz cvičení 4.4.9. Relace $S$ je zřejmě reflexivní a transitivní a je to nejmenší reflexivní a transitivní relace $T$ s vlastností $R \subseteq T$.

V řeči akcí lze reflexivní a transitivní uzávěr popsat takto: akce $v \xrightarrow{S} v'$ znamená existenci konečného (i nulového) počtu navzájem na sebe navazujících akcí $R$ vedoucích z $v$ do $v'$.

Pro libovolnou modální formuli $\varphi$ označme jako $\varphi^*$ následující formuli:

$$\Big([S]\varphi \Rightarrow (\varphi \wedge [R][S]\varphi)\Big) \wedge \Big([S](\varphi \Rightarrow [R]\varphi) \Rightarrow (\varphi \Rightarrow [S]\varphi)\Big)$$

Předpokládejme, že $At \neq \emptyset$. Pro akční strukturu $\mathbb{V}$ typu $\{R, S\}$ jsou následující tvrzení ekvivalentní:

1. Pro libovolné $\varphi$ platí $\models_{\mathbb{V}} \varphi^*$.
2. Relace $S$ je reflexivní a transitivní uzávěr $R$.

1. ⇒ 2. Předpokládejme, že $\models_{\mathbb{V}} \varphi^*$ platí pro libovolné $\varphi$. Potřebujeme ukázat, že pro libovolné vrcholy $v$ a $v'$ platí

$v \xrightarrow{S} v'$ právě tehdy, když existuje $n \geq 0$ tak, že $v \xrightarrow{R^n} v'$.

Důkaz rozdělíme do dvou částí.

(a) Vezměme akci $v \xrightarrow{S} v'$. Chceme ukázat, že existuje nějaké $n \geq 0$ tak, že $v \xrightarrow{R^n} v'$.

Na $\mathbb{V}$ zvolme modální model $\mathscr{M}$ takto:

$$\mathscr{V}(a) = \{w \mid \text{ existuje } n \geq 0 \text{ tak, že } v \xrightarrow{R^n} w\}$$

a pro ostatní atomické formule $b$ je $\mathscr{V}(b) = \emptyset$.

Podle předpokladu platí

$$v \Vdash_{\mathscr{M}} \Big([S]a \Rightarrow (a \wedge [R][S]a)\Big) \wedge \Big([S](a \Rightarrow [R]a) \Rightarrow (a \Rightarrow [S]a)\Big) \tag{5.32}$$

speciálně dostáváme

$$v \Vdash_{\mathscr{M}} [S](a \Rightarrow [R]a) \Rightarrow (a \Rightarrow [S]a) \tag{5.33}$$

Ukážeme nejprve, že platí

$$v \Vdash_{\mathscr{M}} [S](a \Rightarrow [R]a) \tag{5.34}$$

Zvolme tedy libovolnou akci $v \xrightarrow{S} t$ a ukážeme, že $t \Vdash_{\mathscr{M}} a \Rightarrow [R]a$. To je ale triviální, pokud $t \Vdash_{\mathscr{M}} a$ a $t \xrightarrow{R} t'$, znamená to, že do vrcholu $t$ se lze dostat z $v$ několikerým opakováním akce $R$. Tudíž i do vrcholu $t'$ se lze dostat z vrcholu $v$ několikerým opakováním akce $R$. To znamená přesně tolik, že $t' \Vdash_{\mathscr{M}} a$. Tudíž (5.34) platí.

Z platnosti (5.34) a (5.33) však dostáváme

$$v \Vdash_{\mathscr{M}} a \Rightarrow [S]a \tag{5.35}$$

Nejprve nahlédněme, že platí $v \Vdash_{\mathscr{M}} a$. To proto, že z $v$ do $v$ se lze dostat nulovým počtem akcí $R$. Protože platí (5.35), dostáváme platnost

$$v \Vdash_{\mathscr{M}} [S]a \tag{5.36}$$

Připomeňme, že jsme začali akcí $v \xrightarrow{S} v'$. Podle (5.36) tedy víme, že platí $v' \Vdash_{\mathscr{M}} a$, neboli $v' \in \mathscr{V}(a)$. To jsme chtěli dokázat.

(b) Vezměme akci $v \xrightarrow{R^n} v'$. Chceme ukázat existenci akce $v \xrightarrow{S} v'$.

Na $\mathbb{V}$ zvolme modální model $\mathscr{M}$ takto:

$$\mathscr{V}(a) = \{w \mid \text{ existuje akce } v \xrightarrow{S} w\}$$

a pro ostatní atomické formule $b$ je $\mathscr{V}(b) = \emptyset$.

Podle předpokladu platí

$$v \Vdash_{\mathscr{M}} \Big([S]a \Rightarrow (a \wedge [R][S]a)\Big) \wedge \Big([S](a \Rightarrow [R]a) \Rightarrow (a \Rightarrow [S]a)\Big) \tag{5.37}$$

speciálně dostáváme speciálně dostáváme

$$v \Vdash_{\mathscr{M}} [S]a \Rightarrow (a \wedge [R][S]a) \tag{5.38}$$

Podle definice valuace víme, že platí $v \Vdash_{\mathscr{M}} [S]a$, což spolu s (5.38) dává

$$v \Vdash_{\mathscr{M}} a \wedge [R][S]a \tag{5.39}$$

Připomeňme, že máme k dispozici $n$ takové, že $v \xrightarrow{R^n} v'$. Mohou nastat dva případy:

$n = 0$. V tomto případě $v = v'$ a z (5.39) dostáváme $v \Vdash_{\mathscr{M}} a$. Ukázali jsme, že $v \in \mathscr{V}(a)$.

$n > 0$. V tomto případě existuje z $v$ do $v'$ cesta akcemi $R$ délky $n$ vedoucí skrze vrcholy $v_1, \ldots, v_{n+1}$, kde $v = v_1$ a $v_{n+1} = v'$. Lze snadno ukázat, že platí

$$v_i \Vdash_{\mathscr{M}} a \wedge [R]Sa$$

pro všechna $i \in \{1, \ldots, n+1\}$. Speciálně pro $i = n+1$ platí $v_{n+1} \Vdash_{\mathscr{M}} a$, a to jsme chtěli ukázat, protože $v_{n+1} = v'$.

2. $\Rightarrow$ 1. Předpokládejme, že $S$ je reflexivní a transitivní uzávěr relace $R$.

Vezměme libovolný modální model $\mathscr{M}$ na $\mathbb{V}$, libovolnou modální formuli $\varphi$ a libovolný vrchol $v$. Chceme ukázat, že platí $v \Vdash_{\mathscr{M}} \varphi^*$. Důkaz rozdělíme na dvě části.

(a) Ukážeme, že platí

$$v \Vdash_{\mathscr{M}} [S]\varphi \Rightarrow (\varphi \wedge [R][S]\varphi) \tag{5.40}$$

Předpokládejme tedy platnost

$$v \Vdash_{\mathscr{M}} [S]\varphi \tag{5.41}$$

a chceme ukázat, že platí

$$v \Vdash_{\mathscr{M}} \varphi \wedge [R][S]\varphi \tag{5.42}$$

Protože $v \xrightarrow{R^0} v$ a protože $S$ je transitivní uzávěr $R$, víme, že $v \xrightarrow{S} v$. Díky (5.41) tedy dostáváme, že platí

$$v \Vdash_{\mathscr{M}} \varphi \tag{5.43}$$

Abychom ukázali, že $v \Vdash_{\mathscr{M}} [R][S]\varphi$, zvolme jakoukoli akci $v \xrightarrow{R} v'$. Chceme ukázat, že platí $v' \Vdash_{\mathscr{M}} [S]\varphi$.

Zvolme tedy opět jakoukoli akci $v' \xrightarrow{S} v''$. Protože $S$ je reflexivní a transitivní uzávěr $R$, znamená to, že z $v'$ do $v''$ se lze dostat $n$ akcemi $R$ za sebou. Z vrcholu $v$ se lze tudíž dostat do $v''$ $n+1$ akcemi $R$. To ale znamená existenci akce $v \xrightarrow{S} v''$. Z platnosti vztahu (5.41) dostáváme, že $v'' \Vdash_{\mathscr{M}} \varphi$. Spolu s (5.43) jsme tedy ukázali, že platí (5.42).

(b) Ukážeme, že platí

$$v \Vdash_{\mathscr{M}} [S](\varphi \Rightarrow [R]\varphi) \Rightarrow (\varphi \Rightarrow [S]\varphi) \tag{5.44}$$

Předpokládejme tedy, že platí

$$v \Vdash_{\mathscr{M}} [S](\varphi \Rightarrow [R]\varphi) \tag{5.45}$$

a ukážeme, že platí

$$v \Vdash_{\mathscr{M}} \varphi \Rightarrow [S]\varphi \tag{5.46}$$

K tomu účelu budeme předpokládat, že platí

$$v \Vdash_{\mathscr{M}} \varphi \tag{5.47}$$

Abychom ukázali (5.46), zvolme libovolnou akci $v \xrightarrow{S} v'$. Potřebujeme ukázat, že $v \Vdash_{\mathscr{M}} \varphi$. Protože $S$ je reflexivní a transitivní uzávěr $R$ znamená to, že z $v$ do $v'$ se lze dostat $n$ po sobě jdoucími akcemi $R$. Nyní indukcí podle $n$ lze snadno ukázat, že $v' \Vdash_{\mathscr{M}} \varphi$.

V dalším se zaměříme na otázky „dědičnosti“ modální definovatenosti, tj. na problémy typu:

Je-li $\mathbb{V}$ prvek modálně definovatelné třídy $\Lambda$, potom i každý faktorový objekt struktury $\mathbb{V}$ je prvkem $\Lambda$.

Podobný problém budeme řešit i pro podobjekty akčních struktur.

**5.3.4 Věta** *Ať $f : \mathbb{V} \twoheadrightarrow \mathbb{W}$ je surjektivní zig-zag morfismus. Potom pro libovolnou modální formuli $\varphi$ platí:*

*je-li $\models_{\mathbb{V}} \varphi$, pak $\models_{\mathbb{W}} \varphi$.*

Důkaz. Předpokládejme, že $\mathscr{M} = \langle \mathbb{W}, \mathscr{W} \rangle$ je libovolný modální model a $w$ je libovolný vrchol. Protože zobrazení $f$ je surjektivní, existuje vrchol $v$ takový, že $f(v) = w$. Podle předpokladu platí $\models_{f^{-1}(\mathscr{M})} \varphi$, speciálně tedy $v \Vdash_{f^{-1}(\mathscr{M})} \varphi$. Podle důsledku 4.3.7 je tedy $f(v) \Vdash_{\mathscr{M}} \varphi$, což jsme chtěli ukázat. ■

Uvedeme nyní příklad vlastnosti, která není modálně definovatelná.

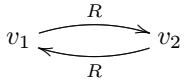
**5.3.5 Definice** Relace $R$ na množině $A$ je *ireflexivní*, pokud pro žádné $x \in A$ neplatí $x \; R \; x$.

**5.3.6 Tvrzení** *Ireflexivita relace $R$ není modálně definovatelná.*

Důkaz. Budeme postupovat sporem. Předpokládejme, že formule $\varphi$ modálně definuje ireflexivitu relace $R$.

Uvažujme o následujících akčních strukturách typu $\{R\}$.

$\mathbb{V}$ je akční struktura




$\mathbb{W}$ je akční struktura

Konstantní zobrazení $f : \{v_1, v_2\} \to \{w\}$ je zřejmě surjektivní zig-zag morfismus $f : \mathbb{V} \twoheadrightarrow \mathbb{W}$. Podle věty 5.3.4 je tedy $\models_{\mathbb{W}} \varphi$, a to je spor. ■

Připomeňme pojmy podobjekt a faktorový objekt z definice 3.3.16. Důsledkem věty 5.3.4 je tedy fakt, že třída akčních struktur modálně definovaných formulí $\varphi$ je uzavřena na zig-zag obrazy. Zformulujeme to přesně:

**5.3.7 Tvrzení** *Ať třída akčních struktur $\Lambda$ je modálně definovaná formulí $\varphi \in Log_{\Delta}(At)$. Jestliže $\mathbb{V} \in \Lambda$ a $\mathbb{W}$ je faktorový objekt struktury $\mathbb{V}$, pak $\mathbb{W} \in \Lambda$.*

Důkaz. Protože $\mathbb{W}$ je faktorový objekt struktury $\mathbb{V}$, existuje surjektivní zig-zag morfismus $f : \mathbb{V} \twoheadrightarrow \mathbb{W}$.

Podle předpokladu je $\models_{\mathbb{V}} \varphi$. Podle věty 5.3.4 je tedy $\models_{\mathbb{W}} \varphi$, neboli $\mathbb{W} \in \Lambda$. ■

Věnujme se nyní otázce uzavřenosti třídy modálně definovatelných struktur na podobjekty.

**5.3.8 Tvrzení** *Ať třída akčních struktur $\Lambda$ je modálně definovaná formulí $\varphi \in Log_{\Delta}(At)$. Jestliže $\mathbb{V} \in \Lambda$ a $\mathbb{W}$ je podobjekt struktury $\mathbb{V}$, pak $\mathbb{W} \in \Lambda$.*

Důkaz. Protože $\mathbb{W}$ je podobjekt struktury $\mathbb{V}$, existuje prostý zig-zag morfismus $f : \mathbb{W} \rightarrowtail \mathbb{V}$.

Stačí ukázat, že $\models_{\mathbb{W}} \varphi$. Vezměme libovolný model $\mathscr{M}$ na $\mathbb{W}$ a libovolný vrchol $w$. Chceme ukázat, že $w \Vdash_{\mathscr{M}} \varphi$.

Protože $f(\mathscr{M})$ je model na $\mathbb{V}$, platí podle předpokladu $\models_{f(\mathscr{M})} \varphi$.

Podle důsledku 4.3.7 platí $w \Vdash_{f^{-1}(f(\mathscr{M}))} \varphi$ právě tehdy, když $f(w) \Vdash_{f(\mathscr{M})} \varphi$. Protože podle lemmatu 4.3.9 je $\mathscr{M} = f^{-1}(f(\mathscr{M}))$, je důkaz hotov. ■

## 5.4 Cvičení

**5.4.1 Cvičení** Rozhodněte, zda pro libovolné modální formule $\varphi$, $\psi$ a akci $R$ platí následující:

1. $\models \langle R\rangle(\varphi \vee \psi) \Rightarrow (\langle R\rangle\varphi \vee \langle R\rangle\psi)$.
2. $\models [R](\varphi \vee \psi) \Rightarrow ([R]\varphi \vee [R]\psi)$.
3. $\models [R]\varphi \Rightarrow \varphi$.
4. $\langle R\rangle\varphi \Rightarrow [R]\langle R\rangle\varphi$.

**5.4.2 Cvičení** Dokažte, že $\mathbf{K} \vdash \varphi \Leftrightarrow \mathtt{tt}$ platí právě tehdy, když platí $\mathbf{K} \vdash \varphi$.

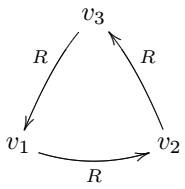
**5.4.3 Cvičení** Pro formuli $\varphi$ označte jako $\varphi[a \mapsto \psi]$ formuli vzniklou z $\varphi$ nahrazením každého výskytu atomické formule $a$ formulí $\psi$.

Ukažte, že pro libovolné $\varphi$ a libovolnou akční strukturu $\mathbb{V}$ jsou následující podmínky ekvivalentní:

1. Platí $\models_{\mathbb{V}} \varphi$.
2. Pro libovolné $\psi$ platí $\models_{\mathbb{V}} \varphi[a \mapsto \psi]$.

**5.4.4 Cvičení** Ať $\Delta = \{R\}$. Ukažte, že $\Lambda G(\varepsilon, R, RR, \varepsilon)$ je přesně třída akčních struktur typu $\Delta$, ve kterých je relace $R$ transitivní.

Předpokládejte, že $At = \emptyset$. Ukažte, že akční struktura $\mathbb{V}$ z následujícího obrázku




splňuje $\models_{\mathbb{V}} [R]\varphi \Rightarrow [RR]\varphi$ pro libovolnou formuli $\varphi$, ačkoli relace $R$ není ve $\mathbb{V}$ transitivní.

**5.4.5 Cvičení** Dejte další příklady modální definovatelnosti pomocí vlastnosti $G(a, b, c, d)$, kde $a$, $b$, $c$, $d$ jsou některé ze symbolů $R$ a $\varepsilon$ podobně, jako v příkladu 5.3.2.

**5.4.6 Cvičení** Řekneme, že akce $R$ je *souvislá* v akční struktuře $\mathbb{V}$, jestliže platí:

Je-li $v \xrightarrow{R} v_1$ a $v \xrightarrow{R} v_2$, potom je $v_1 \xrightarrow{R} v_2$ nebo $v_2 \xrightarrow{R} v_1$.

Rozhodněte, zda následující je ekvivalentní:

1. $R$ je souvislá v akční struktuře $\mathbb{V}$.
2. Pro libovolnou modální formule $\varphi$ a $\psi$ platí $\models_{\mathbb{V}} [R]([R]\varphi \Rightarrow \psi) \vee [R]([R]\psi \Rightarrow \varphi)$.

(Rozlište případy $At \neq \emptyset$ a $At = \emptyset$.)

**5.4.7 Cvičení** Předpokládejte, že $R, S \in \Delta$ a $At \neq \emptyset$. Ukažte, že pro akční strukturu $\mathbb{V}$ je ekvivalentní:

1. Pro libovolné $\varphi$ platí $\models_{\mathbb{V}} [S](\varphi \Rightarrow [R]\varphi) \Rightarrow (\varphi \Rightarrow [S]\varphi)$.
2. Relace $S$ je podmnožinou reflexivního a transitivního uzávěru $R$.

**5.4.8 Cvičení** Předpokládejte, že $R \in \Delta$ a $At \neq \emptyset$. Ukažte, že pro akční strukturu $\mathbb{V}$ je ekvivalentní:

1. Relace $R$ je transitivní ve $\mathbb{V}$ a neexistuje nekonečná posloupnost
$$v_0 \xrightarrow{R} v_1 \xrightarrow{R} v_2 \ldots \xrightarrow{R} v_n \xrightarrow{R} \ldots$$
2. $\models_{\mathbb{V}} [R]([R]\varphi \Rightarrow \varphi) \Rightarrow [R]\varphi$ platí pro libovolnou formuli $\varphi$.

**5.4.9 Cvičení** Ukažte, že následující vlastnost $\mathrm{V}(R)$ akčních struktur není modálně definovatelná.

Akční struktura má vlastnost $\mathrm{V}(R)$ právě tehdy, když v ní existuje vrchol $v$ takový, že $v \xrightarrow{R} v$.

(Návod: použijte tvrzení 5.3.8, tj. nalezněte akční strukturu $\mathbb{V}$ mající vlastnost $\mathrm{V}(R)$ a nalezněte její podstrukturu, která nemá vlastnost $\mathrm{V}(R)$.)

**5.4.10 Cvičení** Připomeňte si definici koproduktu akčních struktur ze cvičení 3.4.8.

Ukažte, že třída akčních struktur modálně definovatelná formulí $\varphi$ je uzavřena na koprodukty. Přesněji: ukažte následující:

Ať $\mathbb{V}_i$, $i \in I$, je systém akčních struktur takových, že pro všechna $i \in I$ platí $\models_{\mathbb{V}_i} \varphi$. Označme jako $\mathbb{V}$ akční strukturu $\coprod_{i \in I} \mathbb{V}_i$. Pak platí $\models_{\mathbb{V}} \varphi$.

**5.4.11 Cvičení** Využijte cvičení 5.4.10 k důkazu toho, že následující vlastnost $\mathrm{V}(R)$ akčních struktur není modálně definovatelná.

Akční struktura má vlastnost $\mathrm{V}(R)$ právě tehdy, když pro libovolné vrcholy $v$ a $w$ existuje akce $v \xrightarrow{R} w$.

Návod: vystačíte s koproduktem dvou akčních struktur.

# Kapitola 6

# Parciální korektnost programů

V této kapitole se seznámíme s aplikací dosavadní teorie na programovací jazyky. Zavedeme jednoduchý imperativní jazyk a jeho zamýšlený modální model. Modální formule nám umožní se formálně vyjádřit o *parciální korektnosti programů*.

Formule parciální korektnosti jsou případem pohledu na program jako na binární relaci na množině podmínek (formulí). Uvažujme o programu $p$ nějakého programovacího jazyka. Dále si představme, že máme k dispozici nějakou množinu formulí $F$ (intuice je zde taková, že dané formule něco říkají o identifikátorech, které program $p$ používá). Program $p$ lze pak chápat jako binární relaci $p : F \multimap F$, kde dvojice formulí $\varphi$, $\psi$ z množiny $F$ je v relaci $p$, pokud platí:

> jestliže před spuštěním programu $p$ platila formule $\varphi$, pak po jeho ukončení platí formule $\psi$.

Aparát modální logiky nám dovolí takové trojice (formule $\varphi$, program $p$, formule $\psi$) elegantně popsat. Jiný způsob popisu těchto trojic nabízí *Hoarova logika* — viz například [RiVe97].

## 6.1 Jazyk `while` programů

V tomto odstavci zavedeme syntaxi takzvaného jazyka `while` programů. Jedná se o jednoduchý imperativní jazyk, který umožňuje základní rekursivní konstrukci neurčité iterace — cyklus `while`. Detaily definice syntaxe tohoto jazyka se nebudeme příliš zabývat. Čtenáře, který chce syntaxi a sémantiku popsat podrobně, odkazujeme buď na cvičení k této kapitole, nebo například na skriptum [RiVe97].

Jazyk `while` programů má čtyři syntaktické typy: `Id` (*identifikátory*), `Comm` (*příkazy*), `AExp` (*aritmetické výrazy*) a `BExp` (*Booleovské výrazy*).

Syntaxe příkazů (někdy jim budeme říkat i *programy*) je dána následovně:

$$p ::= \texttt{skip} \mid x := e \mid p_1; p_2 \mid \texttt{if } b \texttt{ then } p_1 \texttt{ else } p_2 \texttt{ fi} \mid \texttt{while } b \texttt{ do } p \texttt{ od} \tag{6.1}$$

kde $x$ je generické jméno pro identifikátor a $e$ je generické jméno pro aritmetický výraz. Syntaxe Booleovských výrazů je:

$$b ::= e_1 = e_2 \mid e_1 < e_2 \mid b_1 \texttt{ and } b_2 \mid \texttt{not } b \tag{6.2}$$

Syntaxe aritmetických výrazů je:

$$e ::= \texttt{0} \mid \texttt{1} \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 * e_2 \mid x \tag{6.3}$$

kde $x$ je generické jméno pro identifikátor.

Co se týče identifikátorů, předpokládáme, že je dána předem neprázdná spočetná množina, ze které identifikátory vybíráme.

Dále budeme používat, bude-li to zapotřebí, samozřejmé *syntaktické zkratky*; například budeme psát `true` místo `1 = 1`, budeme psát `2` místo `1 + 1` a podobně.

Bude-li to zapotřebí, budeme dále používat i některé syntaktické konstrukce, které jsme nedefinovali. Vždy však vysvětlíme jejich povahu.

## 6.2 Zamýšlený modální model

Zamýšleným modálním modelem rozumíme vlastně sémantiku jazyka z odstavce 6.1. Abychom definovali modální model, je zapotřebí:

1. Definovat modální logiku, ve které chceme pracovat, tj. typ $\Delta$ a množinu atomických formulí $At$.
2. Definovat akční strukturu $\mathbb{V}$ typu $\Delta$, tj. popsat její vrcholy a interpretace akcí.
3. Definovat valuaci $\mathscr{V}$ atomických formulí, tj. popsat, ve kterých vrcholech jsou dané atomické formule vynuceny.

Množina $\Delta$ bude obsahovat jako prvky všechny příkazy a všechny Booleovské výrazy jazyka `while` programů. Připomeňme, že množina $\Delta$ je nekonečná (spočetná) množina, má ale vnitřní strukturu — její prvky jsou induktivně vytvářeny podle syntaktických pravidel pro tvorbu příkazů a Booleovských výrazů. Tohoto faktu využijeme později při definici akcí ve struktuře $\mathbb{V}$.

Množina $At$ bude obsahovat jako prvky všechny zápisy tvaru $e_1 \leq e_2$, kde $e_1$ a $e_2$ jsou aritmetické výrazy jazyka `while` programů. I množina $At$ je tedy spočetná.

Vrcholům akční struktury $\mathbb{V}$ budeme říkat *paměti* a typický vrchol budeme v této kapitole značit $M$. Pamětí budeme rozumět seznam všech dvojic (identifikátor, jeho hodnota). Než budeme moci paměť přesně definovat, učiníme následující předpoklady:

1. Množina `Id` všech identifikátorů je předem daná spočetná množina.
2. Vezměme množinu $\mathbb{Z}$ všech celých čísel (spolu s obvyklými algebraickými operacemi) a přidejme k ní prvek $\perp_{\mathbb{Z}}$, který není celým číslem. Prvek $\perp_{\mathbb{Z}}$ budeme číst: *nedefinováno v* $\mathbb{Z}$. Množině $\mathbb{Z} \cup \{\perp_{\mathbb{Z}}\}$ budeme říkat *množina hodnot identifikátorů*.

*Paměť* $M$ je pak množina dvojic $(x, v)$, kde $x$ je identifikátor a $v$ je prvek množiny hodnot identifikátorů (tj. $v$ je buď celé číslo nebo $\perp_{\mathbb{Z}}$). V paměti $M$ musí být každý identifikátor zastoupen právě jednou. To lze říci přesněji tak, že $M$ je funkce z množiny `Id` do množiny $\mathbb{Z} \cup \{\perp_{\mathbb{Z}}\}$.

Pro identifikátor $x$ a hodnotu $v$ zavedeme *update paměti* $M$ *o hodnotu* $v$ *pro* $x$ jako paměť, značenou $M[x \mapsto v]$, následovně: změníme funkční hodnotu $M$ v bodě $x$ tak, aby byla rovna $v$.

Je-li dána paměť $M$, budeme říkat, že *identifikátor* $x$ *má* $v$ *paměti* $M$ *hodnotu* $v$, jestliže $(x, v) \in M$. Tato definice umožňuje v paměti $M$ vyhodnotit každý aritmetický výraz jako nějaké celé číslo nebo hodnotu $\perp_{\mathbb{Z}}$ a každý Booleovský výraz jako pravdivý, nepravdivý nebo nedefinovaný — viz cvičení 6.5.1.

Nyní následuje definice akcí ve struktuře $\mathbb{V}$. Akce budeme definovat induktivně následujícími odvozovacími pravidly:

$$\frac{}{M \xrightarrow{b} M} \qquad \text{je-li } b \text{ pravdivé v paměti } M \tag{6.4}$$

Toto pravidlo je třeba chápat jako axiom: na to, abychom odvodili akci $M \xrightarrow{b} M$ je třeba pouze vědět, že Booleovský výraz $b$ je pravdivý v paměti $M$.

Pravidlo pro příkaz `skip`

$$\frac{}{M \xrightarrow{\texttt{skip}} M} \tag{6.5}$$

je také axiom, stejně jako pravidlo pro dosazovací příkaz:

$$\frac{}{M \xrightarrow{x := e} M[x \mapsto v]} \qquad \text{kde } v \text{ je hodnota } e \text{ v paměti } M \tag{6.6}$$

Další pravidla jsou odvozovací. Nejprve pravidlo pro zřetězování příkazů:

$$\frac{M_1 \xrightarrow{p} M' \qquad M' \xrightarrow{q} M_2}{M_1 \xrightarrow{p\,;\,q} M_2} \tag{6.7}$$

Pravidla pro podmíněný příkaz jsou dvě. První vystihuje situaci provádění „then“ větve podmíněného příkazu

$$\frac{M_1 \xrightarrow{b} M_1 \qquad M_1 \xrightarrow{p_1} M_2}{M_1 \xrightarrow{\texttt{if } b \texttt{ then } p_1 \texttt{ else } p_2 \texttt{ fi}} M_2} \tag{6.8}$$

druhé provádění „else“ větve podmíněného příkazu

$$\frac{M_1 \xrightarrow{\texttt{not } b} M_1 \qquad M_1 \xrightarrow{p_2} M_2}{M_1 \xrightarrow{\texttt{if } b \texttt{ then } p_1 \texttt{ else } p_2 \texttt{ fi}} M_2} \tag{6.9}$$

První pravidlo pro konstrukci while cyklu vystihuje situaci, kdy cyklus vůbec neproběhne

$$\frac{M \xrightarrow{\texttt{not } b} M}{M \xrightarrow{\texttt{while } b \texttt{ do } p \texttt{ od}} M} \tag{6.10}$$

druhé pravidlo zavádí rekursivní chování cyklu `while` $b$ `do` $p$ `od`

$$\frac{M_1 \xrightarrow{b} M_1 \qquad M_1 \xrightarrow{p\,;\,\texttt{while } b \texttt{ do } p \texttt{ od}} M_2}{M_1 \xrightarrow{\texttt{while } b \texttt{ do } p \texttt{ od}} M_2} \tag{6.11}$$

Definice akční struktury $\mathbb{V}$ je tímto ukončena.

K definici modálního modelu $\mathscr{M}$ nyní zbývá pouze dát valuaci atomických proměnných. Připomeňme, že typická atomická formule je tvaru $e_1 \leq e_2$, kde $e_1$ a $e_2$ jsou aritmetické výrazy jazyka `while` programů. Množinu $\mathscr{V}(e_1 \leq e_2)$ budeme definovat jako množinu všech pamětí $M$, ve kterých je Booleovský výraz $e_1 \leq e_2$ pravdivý — viz cvičení 6.5.3.

Výsledný modální model nazveme $\mathscr{M}$ a *v této kapitole o jiném modálním modelu hovořit nebudeme.*

**6.2.1 Konvence** Zaveďme několik konvencí, které nám umožní snazší vyjadřování.

1. V paměti budeme vypisovat pouze ty identifikátory, které mají jako hodnotu celé číslo. To znamená, že budeme psát například $\{(x, 1), (y, -2)\}$ pro paměť $M$, kde $(z, \perp_{\mathbb{Z}}) \in M$ pro všechny identifikátory $z$ různé od $x$ a $y$.

2. Pro formuli připustíme například i zápis $e_1 = e_2$ jako zkratku za konjunkci dvou atomických formulí $(e_1 \leq e_2) \wedge (e_2 \leq e_1)$. Podobné zápisy, které budeme používat, jsou například $e_1 < e_2$, $e_1 \geq e_2$.

## 6.3 Formule parciální korektnosti

Zamýšlený modální model nám umožní precisovat úvahy z úvodu této kapitoly.

**6.3.1 Definice** *Formule parciální korektnosti programu* $p$ je formule $\varphi \Rightarrow [p]\psi$, kde formule $\varphi$ a $\psi$ neobsahují modality. V tomto kontextu budeme formuli $\varphi$ říkat *prepodmínka* a formuli $\psi$ budeme říkat *postpodmínka* (programu $p$).

**6.3.2 Příklad** Ukážeme, že platí $\models_{\mathscr{M}} ((n \geq 0) \wedge (x \geq 0)) \Rightarrow [p]y = x^n$, kde $p$ je následující program

$$k := n\,;\; y := 1\,;\; z := x;$$
$$\texttt{while } k > 0 \texttt{ do if odd } k \texttt{ then } k := k - 1\,;\; y := y * z \texttt{ else } k := k/2\,;\; z := z * z \texttt{ fi od}$$

kde `odd` $k$ je test na lichost čísla $k$ a $k/2$ je celočíselné dělení čísla $k$ číslem 2.

Označme preambuli programu $p$, tj. sadu dosazovacích příkazů $k := n\,;\, y := 1\,;\, z := x$ jako $a$, část obsahující cyklus `while` jako $w$ a část s podmíněným příkazem (tělo cyklu `while`) jako $q$.

Abychom ukázali platnost $\models_{\mathscr{M}} ((n \geq 0) \wedge (x \geq 0)) \Rightarrow [p]y = x^n$, ukážeme platnost pomocného tvrzení $T(n)$ pro všechna $n \geq 0$.

$T(n)$ je následující tvrzení:

> Pro všechna $K$, $l \geq 0$ a $m \geq 0$ platí:
>
> > jestliže $K \Vdash_{\mathscr{M}} (k = n) \wedge (y = x^l) \wedge (z = x^m)$ a $K \xrightarrow{w} K'$, pak platí $K' \Vdash_{\mathscr{M}} y = k^{l+m\cdot n}$.

Jakmile bude platnost $T(n)$ pro všechna $n \geq 0$ dokázána, ukážeme platnost

$$\models_{\mathscr{M}} ((n \geq 0) \wedge (x \geq 0)) \Rightarrow [p]y = x^n$$

následovně: zvolme jakoukoli paměť $M$ takovou, že $M \Vdash (n \geq 0) \wedge (x \geq 0)$ a $M \xrightarrow{p} M'$. Podle odvozovacích pravidel pro akce víme, že existuje paměť $N$ taková, že $M \xrightarrow{a} N$ a $N \xrightarrow{w} M'$. Navíc zřejmě $N \Vdash_{\mathscr{M}} (k = n) \wedge (y = x^0) \wedge (z = x^1)$. Podle tvrzení $T(n)$ potom $M' \Vdash_{\mathscr{M}} y = k^{0+1\cdot n}$, což jsme měli ukázat.

Důkaz platnosti $T(n)$ pro všechna $n \geq 0$ povedeme matematickou indukcí podle $n$.

- $n = 0$. Podle pravidel pro tvorbu akcí cyklus `while` vůbec neproběhne. Platí tedy $K = K'$ a zřejmě $K' \Vdash_{\mathscr{M}} y = k^{l+m\cdot 0}$.
- Předpokládejme, že $n > 0$ a že tvrzení $T(n')$ platí pro všechna $n' \leq n$. Dále předpokládejme, že máme paměť $K$ takovou, že $K \Vdash_{\mathscr{M}} (k = n+1) \wedge (y = x^l) \wedge (z = x^m)$ a $K \xrightarrow{w} K'$. Díky tomu, že $n + 1 > 0$, cyklus `while` proběhne alespoň jednou. Existuje tedy paměť $N$ taková, že $K \xrightarrow{q} N$ a $N \xrightarrow{w} K'$.

  Rozlišíme dva případy.

  > Číslo $n + 1$ je liché. Potom zřejmě $N \Vdash_{\mathscr{M}} (k = n) \wedge (y = x^{l+m}) \wedge (z = x^m)$. Podle indukčního předpokladu použitého na paměť $N$ je tedy $K' \Vdash_{\mathscr{M}} y = k^{l+m+m\cdot n}$, což jsme chtěli dokázat.
  >
  > Číslo $n + 1$ je sudé. Potom zřejmě $N \Vdash_{\mathscr{M}} (k = \frac{n+1}{2}) \wedge (y = x^l) \wedge (z = x^{2m})$. Podle indukčního předpokladu použitého na paměť $N$ je tedy $K' \Vdash_{\mathscr{M}} y = k^{l+2m\cdot(\frac{n+1}{2})}$, což jsme chtěli dokázat.

Viz také cvičení 6.5.7.

## 6.4 Deduktivní systém formulí parciální korektnosti

Víme, že platnost formule parciální korektnosti $\varphi \Rightarrow [p]\psi$ programu $p$ v zamýšleném modelu $\mathscr{M}$ přesně vystihuje situaci popsanou v úvodu k této kapitole, tj. $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$ platí právě tehdy, když pro každou paměť $M$ platí:

> jestliže $M \Vdash_{\mathscr{M}} \varphi$ a $M \xrightarrow{p} M'$, pak $M' \Vdash_{\mathscr{M}} \psi$.

V tomto smyslu je možno chápat dvojici formulí $\varphi$ a $\psi$ takovou, že $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$, jako *specifikaci* chování programu $p$. V příkladu 6.3.2 jsme viděli, že důkaz platnosti $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$ nemusí být jednoduchý. V tomto odstavci zavedeme systém odvozovacích pravidel, který nám umožní formule takové, že platí vztah $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$, popsat *syntakticky*. Tento deduktivní systém pro formule parciální korektnosti nazveme **T** a budeme mu říkat *systém úspěšných tabel pro formule parciální korektnosti*.

Deduktivní systém **T** bude samozřejmě obsahovat axiomy a odvozovací pravidla. Tato pravidla však budeme psát obráceně, než jsme to doposud dělali. Tento přístup (takzvané *Goal Directed Reasoning*), je v literatuře o programovacích jazycích běžný.

Pravidlo

$$\frac{Z}{P_1 \quad P_2 \quad \ldots \quad P_n}$$

budeme číst jako odvozovací pravidlo, ale „vzhůru nohama“: na to, abychom úspěšně odvodili $Z$, je třeba úspěšně odvodit $P_1$, $P_2$, …, $P_n$.

Dále budeme předpokládat, že pro libovolné formule $\varphi$ a $\psi$ bez modalit umíme vždy rozhodnout pravdivost tvrzení $\models_{\mathscr{M}} \varphi \Rightarrow \psi$.

V následujících odvozovacích pravidlech jsou všechny formule $\alpha$, $\beta$ bez modalit. Obsahuje-li formule $\beta$ proměnnou $x$, pak $\beta[x \mapsto e]$ značí formuli vzniklou z $\beta$ dosazením hodnoty $e$ za $x$.

$$\frac{\alpha \Rightarrow [x := e]\beta}{\alpha \Rightarrow \beta[x \mapsto e]} \ (:=)$$

$$\frac{\alpha \Rightarrow [\texttt{skip}]\beta}{\alpha \Rightarrow \beta} \ (\text{skip})$$

$$\frac{\alpha \Rightarrow [p\,;q]\beta}{\alpha \Rightarrow [p]\gamma \quad \gamma \Rightarrow [q]\beta} \ (;)$$

$$\frac{\alpha \Rightarrow [\texttt{if } b \texttt{ then } p_1 \texttt{ else } p_2 \texttt{ fi}]\beta}{(\alpha \wedge b) \Rightarrow [p_1]\beta \quad (\alpha \wedge \neg b) \Rightarrow [p_2]\beta} \ (\text{if})$$

$$\frac{\alpha \Rightarrow [\texttt{while } b \texttt{ do } p \texttt{ od}](\alpha \wedge \neg b)}{(\alpha \wedge b) \Rightarrow [p]\alpha} \ (\text{while})$$

$$\frac{\alpha \Rightarrow [p]\beta}{\alpha_1 \Rightarrow [p]\beta_1} \ (\text{oslabení}) \quad \text{je-li } \models_{\mathscr{M}} \alpha \Rightarrow \alpha_1 \text{ a } \models_{\mathscr{M}} \beta_1 \Rightarrow \beta$$

*Tablo v systému* **T** je vlastně důkaz v deduktivním systému psaný „vzhůru nohama“. Díky výše popsaným pravidlům je zřejmé, že listy tabla mohou být pouze formule typu $\varphi \Rightarrow \psi$, kde $\varphi$ a $\psi$ neobsahují žádné modality. List tabla tvaru $\varphi \Rightarrow \psi$ nazveme *úspěšný*, pokud platí $\models_{\mathscr{M}} \varphi \Rightarrow \psi$. *Tablo je úspěšné*, pokud každý jeho list je úspěšný. Fakt, že formule parciální korektnosti $\varphi \Rightarrow [p]\psi$ má úspěšné tablo, budeme značit $\mathbf{T} \vdash \varphi \Rightarrow [p]\psi$.

**6.4.1 Příklad** Označme program $y := 1\,;$ `while` $x \geq 1$ `do` $y := y * x\,;\ x := x - 1$ `od` jako $p$, `while` cyklus v něm obsažený jako $w$, tělo cyklu $w$ jako $q$.

Následující tablo je úspěšné pro formuli $(x = n \wedge n \geq 0) \Rightarrow [p](y = n!)$.

$$\dfrac{\dfrac{\dfrac{(x = n \wedge n \geq 0) \Rightarrow [p](y = n!)}{\dfrac{(x = n \wedge n \geq 0) \Rightarrow [y := 1](x! * y = n!)}{(x = n \wedge n \geq 0) \Rightarrow (x! = n!)}} \quad \dfrac{(x! * y = n!) \Rightarrow [w](y = n!)}{\dfrac{(x! * y = n!) \Rightarrow [w](x! * y = n! \wedge \neg(x \geq 1))}{(x! * y = n! \wedge x \geq 1) \Rightarrow [q](x! * y = n!)}}}{\dfrac{(x! * y = n! \wedge x \geq 1) \Rightarrow [y := x * y]\varphi}{(x! * y = n! \wedge x \geq 1) \Rightarrow \varphi[y \mapsto x * y]}} \quad \dfrac{\varphi \Rightarrow [x := x - 1](x! * y = n!)}{\varphi \Rightarrow \varphi}}$$

kde formuli $(x - 1)! * y = n!$ jsme označili jako $\varphi$. Viz cvičení 6.5.12.

Při konstrukci úspěšného tabla pro program, který obsahuje cyklus `while` je nejobtížnější nalézt formuli bez modalit, jejíž platnost se nezmění při průchodu cyklem — takzvaný *invariant*. Následující pojem nám existenci invariantu vždy zaručí; problém však bude v tom, že tato existence bude nekonstruktivního charakteru.

**6.4.2 Definice** Ať $\varphi$ je formule bez modalit a ať $p$ je program jazyka `while` programů. Formuli bez modalit $\mathsf{wlp}(\varphi \mid p)$ budeme říkat *nejslabší liberální prepodmínka programu $p$ vzhledem k $\varphi$*, pokud pro každou paměť $M$ platí

$$M \Vdash_{\mathscr{M}} \mathsf{wlp}(\varphi \mid p) \text{ právě tehdy, když } M \Vdash_{\mathscr{M}} [p]\varphi.$$

**6.4.3 Poznámka** Nejslabší liberální prepodmínku lze chápat následovně: formule $[p]\varphi$ rozděluje množinu všech pamětí na dva typy pamětí: na ty paměti, ve kterých je formule $[p]\varphi$ $\mathscr{M}$-vynucena a na ty, ve kterých $\mathscr{M}$-vynucena není.

V tomto smyslu lze chápat formuli $[p]\varphi$ jako unární predikát na množině všech pamětí. Nejslabší liberální prepodmínka (pokud existuje) pak dává možnost popsat tento predikát formulí bez modalit — srovnejte s definicí extense formule 7.2.1.

Následující lemma vysvětluje termín nejslabší liberální prepodmínka (viz také cvičení 6.5.13).

**6.4.4 Lemma** *Ať $p$ je libovolný program, $\varphi$ a $\psi$ ať jsou formule bez modalit. Předpokládejme, že formule* $\mathsf{wlp}(\psi \mid p)$ *existuje. Pak platí:*

1. $\models_{\mathscr{M}} \mathsf{wlp}(\psi \mid p) \Rightarrow [p]\psi$.
2. *Jestliže platí* $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$, *pak* $\models_{\mathscr{M}} \varphi \Rightarrow \mathsf{wlp}(\psi \mid p)$.

Důkaz. 1. Máme-li paměť $M$ takovou, že $M \Vdash_{\mathscr{M}} \mathsf{wlp}(\psi \mid p)$ a je-li $M \xrightarrow{p} M'$, pak podle definice nejslabší liberální prepodmínky platí $M' \Vdash_{\mathscr{M}} \psi$.

2. Předpokládejme, že platí $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$ a zvolme paměť $M$ takovou, že $M \Vdash_{\mathscr{M}} \varphi$. Podle předpokladu $M \Vdash_{\mathscr{M}} [p]\psi$ a tudíž $M \Vdash_{\mathscr{M}} \mathsf{wlp}(\psi \mid p)$ podle definice nejslabší liberální prepodmínky. ■

Nalezněme nyní nejslabší liberální prepodmínky pro ty programy jazyka `while`, které neobsahují cyklus `while`.

**6.4.5 Tvrzení** *Předpokládejme, že $\varphi$ je formule bez modalit a že $p$ je program, který neobsahuje cyklus* `while`. *Potom* $\mathsf{wlp}(\varphi \mid p)$ *vždy existuje.*

Důkaz. Budeme postupovat indukcí podle syntaktické složitosti programu $p$.

1. $p$ je $x := e$. Potom $\mathsf{wlp}(\varphi \mid p)$ je formule $\varphi[x \mapsto e]$.
2. $p$ je `skip`. Potom $\mathsf{wlp}(\varphi \mid p)$ je formule $\varphi$.
3. $p$ je $p_1 ; p_2$. Potom $\mathsf{wlp}(\varphi \mid p)$ je formule $\mathsf{wlp}(\mathsf{wlp}(\varphi \mid p_2) \mid p_1)$.
4. $p$ je `if` $b$ `then` $p_1$ `else` $p_2$ `fi`. Potom $\mathsf{wlp}(\varphi \mid p)$ je formule $(b \wedge \mathsf{wlp}(\varphi \mid p_1)) \vee (\neg b \wedge \mathsf{wlp}(\varphi \mid p_2))$.

Ověření přenecháváme jako cvičení 6.5.14 ■

**6.4.6 Věta Věta o korektnosti pro parciální korektnost.** *Pro libovolné formule $\alpha$ a $\beta$ bez modalit a libovolný program $p$ jazyka* `while` *programů platí:*

*Jestliže* $\mathbf{T} \vdash \alpha \Rightarrow [p]\beta$, *pak* $\models_{\mathscr{M}} \alpha \Rightarrow [p]\beta$.

Důkaz. Stačí ukázat, že všechna pravidla deduktivního systému **T** ze strany 48 jsou korektní.

1. Pravidlo (:=). Předpokládejme platnost vztahu $\models_{\mathscr{M}} \alpha \Rightarrow \beta[x \mapsto v]$. Chceme ukázat, že platí vztah $\models_{\mathscr{M}} \alpha \Rightarrow [x := e]\beta$.

   Zvolme $M$ tak, že $M \Vdash_{\mathscr{M}} \alpha$ a libovolnou akci $x := e$ z paměti $M$ do paměti $M'$. Podle pravidel pro tvorbu akcí je $M' = M[x \mapsto e]$. Dále podle našeho předpokladu platí $M \Vdash_{\mathscr{M}} \beta[x \mapsto e]$. Platnost $M \Vdash_{\mathscr{M}} \beta[x \mapsto e]$ je ale rovnocenná platnosti $M[x \mapsto e] \Vdash_{\mathscr{M}} \beta$. Ukázali jsme, že $M' \Vdash_{\mathscr{M}} \beta$.

2. Pravidlo (skip). Předpokládejme, že platí $\models_{\mathscr{M}} \alpha \Rightarrow \beta$.

   Zvolme $M$ tak, že $M \Vdash_{\mathscr{M}} \alpha$ a libovolnou akci `skip` z paměti $M$ do paměti $M'$. Podle pravidel pro tvorbu akcí je $M' = M$. Ukázali jsme, že $M' \Vdash_{\mathscr{M}} \beta$.

3. Pravidlo (;). Předpokládejme, že platí $\models_{\mathscr{M}} \alpha \Rightarrow [p]\gamma$ a současně $\models_{\mathscr{M}} \gamma \Rightarrow [q]\beta$.

   Zvolme $M$ tak, že $M \Vdash_{\mathscr{M}} \alpha$ a libovolnou akci $p ; q$ z paměti $M$ do paměti $M'$. Podle pravidel pro tvorbu akcí musí existovat paměť $M_1$ taková, že je akce $p$ z $M$ do $M_1$ a akce $q$ z $M_1$ do $M'$. Podle předpokladů ale $M_1 \Vdash_{\mathscr{M}} \gamma$ a opět podle předpokladů $M' \Vdash_{\mathscr{M}} \beta$.

4. Pravidlo (if). Předpokládejme, že platí $\models_{\mathscr{M}} (\alpha \wedge b) \Rightarrow [p_1]\beta$ a současně $\models_{\mathscr{M}} (\alpha \wedge \neg b) \Rightarrow [p_2]\beta$.

   Zvolme $M$ tak, že $M \Vdash_{\mathscr{M}} \alpha$ a libovolnou akci `if` $b$ `then` $p_1$ `else` $p_2$ `fi` z paměti $M$ do paměti $M'$. Rozlišíme dva případy:

$M \Vdash_{\mathscr{M}} b$. Potom podle pravidel pro tvorbu akcí musí existovat paměť $M_1$ taková, že je akce $p_1$ z $M$ do $M_1$. Protože platí $M \Vdash_{\mathscr{M}} (b \wedge \alpha)$, je podle předpokladů $M_1 \Vdash_{\mathscr{M}} \beta$.

$M \Vdash_{\mathscr{M}} \neg b$. Potom podle pravidel pro tvorbu akcí musí existovat paměť $M_1$ taková, že je akce $p_2$ z $M$ do $M_1$. Protože platí $M \Vdash_{\mathscr{M}} (\neg b \wedge \alpha)$, je podle předpokladů $M_1 \Vdash_{\mathscr{M}} \beta$.

5. Pravidlo (while). Předpokládejme, že platí $\models_{\mathscr{M}} (\alpha \wedge b) \Rightarrow [p]\alpha$.

   Zvolme $M$ tak, že $M \Vdash_{\mathscr{M}} \alpha$ a libovolnou akci `while` $b$ `do` $p$ `od` z paměti $M$ do paměti $M'$.

   Budeme postupovat indukcí podle výšky stromu, který akci `while` $b$ `do` $p$ `od` z paměti $M$ do paměti $M'$ odvozuje.

   - Výška odvozovacího stromu je 1. To znamená, že $M \Vdash_{\mathscr{M}} \neg b$ a $M = M'$. Ukázali jsme, že $M' \Vdash_{\mathscr{M}} (\neg b \wedge \alpha)$.
   - Výška odvozovacího stromu akce `while` $b$ `do` $p$ `od` z paměti $M$ do paměti $M'$ je $n > 1$ a předpokládejme, že tvrzení platí pro všechny akce `while` $b$ `do` $p$ `od` s odvozovacím stromem výšky menší než $n$.

     Náš předpoklad o odvozovacím stromu říká, že jsme v situaci, kdy $M \Vdash_{\mathscr{M}} b$ a existuje paměť $M_1$ taková, že je akce $p$ z $M$ do $M_1$ a akce `while` $b$ `do` $p$ `od` z $M_1$ do $M'$. Protože platí $M \Vdash_{\mathscr{M}} (b \wedge \alpha)$, je podle předpokladů $M_1 \Vdash_{\mathscr{M}} \alpha$.

     Akce `while` $b$ `do` $p$ `od` z $M_1$ do $M'$ má odvozovací strom výšky menší než $n$ a podle indukčního předpokladu je $M' \Vdash_{\mathscr{M}} (\neg b \wedge \alpha)$.

■

Větu o úplnosti jsme shopni dokázat za předpokladu, že umíme pro každý program nalézt nejslabší liberální prepodmínku. Podle tvrzení 6.4.5 to lze udělat algoritmicky pro libovolný program neobsahující cyklus `while`.

**6.4.7 Věta Věta o úplnosti pro parciální korektnost.** *Předpokládejme, že pro každou dvojici $p$ a $\varphi$ existuje nejslabší liberální prepodmínka* $\mathsf{wlp}(\varphi \mid p)$. *Potom pro libovolné formule $\alpha$ a $\beta$ bez modalit a libovolný program $p$ jazyka* `while` *programů platí:*

*Jestliže $\models_{\mathscr{M}} \alpha \Rightarrow [p]\beta$, pak $\mathbf{T} \vdash \alpha \Rightarrow [p]\beta$.*

Důkaz. Budeme postupovat indukcí podle syntaktické složitosti programu $p$. Pro $p$ dáme návod, jak sestrojit úspěšné tablo. Vždy budeme předpokládat, že platí $\models_{\mathscr{M}} \alpha \Rightarrow [p]\beta$.

1. $p$ je $x := e$. Tablo
$$\frac{\alpha \Rightarrow [x := e]\beta}{\alpha \Rightarrow \beta[x \mapsto e]} \ (:=)$$
   je úspěšné, protože předpokládáme $\models_{\mathscr{M}} \alpha \Rightarrow [x := e]\beta$.

2. $p$ je `skip`. Tablo
$$\frac{\alpha \Rightarrow [\texttt{skip}]\beta}{\alpha \Rightarrow \beta} \ (\text{skip})$$
   je úspěšné, protože předpokládáme $\models_{\mathscr{M}} \alpha \Rightarrow [\texttt{skip}]\beta$.

3. $p$ je $p\,;\,q$. Jako indukční předpoklad předpokládejme existenci úspěšných tabel pro $\alpha_p \Rightarrow [p]\beta_p$ a $\alpha_q \Rightarrow [q]\beta_q$, kde $\alpha_p$, $\alpha_q$, $\beta_p$ a $\beta_q$ jsou libovolné formule bez modalit takové, že platí $\models_{\mathscr{M}} \alpha_p \Rightarrow [p]\beta_p$ a $\models_{\mathscr{M}} \alpha_q \Rightarrow [q]\beta_q$.

   Protože předpokládáme, že existuje formule $\mathsf{wlp}(\beta \mid p\,;\,q)$ a protože platí $\models_{\mathscr{M}} \alpha \Rightarrow [p\,;\,q]\beta$, platí podle lemmatu 6.4.4 $\models_{\mathscr{M}} \alpha \Rightarrow \mathsf{wlp}(\beta \mid p\,;\,q)$. Můžeme tedy použít pravidlo (oslabení) a začátek úspěšného tabla bude
$$\frac{\alpha \Rightarrow [p\,;\,q]\beta}{\mathsf{wlp}(\beta \mid p\,;\,q) \Rightarrow [p\,;\,q]\beta} \ (\text{oslabení})$$
   Stačí tedy sestrojit úspěšné tablo pro formuli $\mathsf{wlp}(\beta \mid p\,;\,q) \Rightarrow [p\,;\,q]\beta$.

   Podle předpokladů existuje formule $\mathsf{wlp}(\beta \mid q)$. Ukážeme, že:

(a) Formule $\mathsf{wlp}(\beta \mid p\,;\, q) \Rightarrow [p]\mathsf{wlp}(\beta \mid q)$ má úspěšné tablo. K tomu je zapotřebí ukázat, že platí

$$\models_{\mathscr{M}} \mathsf{wlp}(\beta \mid p\,;\, q) \Rightarrow [p]\mathsf{wlp}(\beta \mid q) \tag{6.12}$$

a využít pak indukční předpoklad.
Podle tvrzení 6.4.5 je $\mathsf{wlp}(\beta \mid p\,;\, q)$ formule $\mathsf{wlp}(\mathsf{wlp}(\beta \mid q) \mid p)$.
Předpokládejme, že paměť $M$ je taková, že

$$M \Vdash_{\mathscr{M}} \mathsf{wlp}(\beta \mid p\,;\, q) \tag{6.13}$$

a že máme akci $p$ z $M$ do $M'$. Chceme ukázat, že platí

$$M' \Vdash_{\mathscr{M}} \mathsf{wlp}(\beta \mid q). \tag{6.14}$$

K tomu (podle definice nejslabší liberální prepodmínky) stačí ukázat, že

$$M' \Vdash_{\mathscr{M}} [q]\beta. \tag{6.15}$$

Vezměme tedy libovolnou akci $q$ z $M'$ do $M_1$. Potom ale $M_1 \Vdash_{\mathscr{M}} \beta$, protože podle definice nejslabší liberální predpodmínky vztah 6.13 platí právě tehdy, když platí $M \Vdash_{\mathscr{M}} [p\,;\, q]\beta$.

(b) Formule $\mathsf{wlp}(\beta \mid q) \Rightarrow [q]\beta$ má úspěšné tablo. K tomu je zapotřebí ukázat, že platí

$$\models_{\mathscr{M}} \mathsf{wlp}(\beta \mid q) \Rightarrow [q]\beta \tag{6.16}$$

a využít pak indukční předpoklad. Platnost (6.16) je však jedna z vlastností nejslabší liberální predpodmínky (viz vlastnost 1. z lemmatu 6.4.4).

Začátek úspěšného tabla je:

$$\dfrac{\dfrac{\alpha \Rightarrow [p\,;\, q]\beta}{\mathsf{wlp}(\beta \mid p\,;\, q) \Rightarrow [p\,;\, q]\beta}\ (\text{oslabení})}{\mathsf{wlp}(\beta \mid p\,;\, q) \Rightarrow [p]\mathsf{wlp}(\beta \mid q) \qquad \mathsf{wlp}(\beta \mid q) \Rightarrow [q]\beta}\ (;)$$

4. $p$ je `if` $b$ `then` $p_1$ `else` $p_2$ `fi`. Jako indukční předpoklad předpokládejme existenci úspěšných tabel pro $\alpha_1 \Rightarrow [p_1]\beta_1$ a $\alpha_2 \Rightarrow [p_2]\beta_2$, kde $\alpha_1$, $\alpha_2$, $\beta_1$ a $\beta_2$ jsou libovolné formule bez modalit takové, že platí $\models_{\mathscr{M}} \alpha_1 \Rightarrow [p_1]\beta_1$ a $\models_{\mathscr{M}} \alpha_2 \Rightarrow [p_2]\beta_2$.

Ukážeme, že začátek úspěšného tabla bude

$$\dfrac{\alpha \Rightarrow [\texttt{if}\ b\ \texttt{then}\ p_1\ \texttt{else}\ p_2\ \texttt{fi}]\beta}{(\alpha \wedge b) \Rightarrow [p_1]\beta \qquad (\alpha \wedge \neg b) \Rightarrow [p_2]\beta}\ (\text{if})$$

K tomu stačí ukázat, že platí

$$\models_{\mathscr{M}} (\alpha \wedge b) \Rightarrow [p_1]\beta \tag{6.17}$$

a

$$\models_{\mathscr{M}} (\alpha \wedge \neg b) \Rightarrow [p_2]\beta \tag{6.18}$$

a pak použít indukční přepoklad. Ukážeme platnost (6.17), platnost (6.18) se ukáže analogicky.

Přepokládejme, že $M \Vdash_{\mathscr{M}} \alpha \wedge b$ a že $p_1$ je akce z $M$ do $M'$. Podle pravidel pro tvorbu akcí je pak `if` $b$ `then` $p_1$ `else` $p_2$ `fi` akce z $M$ do $M'$ a tudíž $M' \Vdash_{\mathscr{M}} \beta$, protože jsme předpokládali

$$\models_{\mathscr{M}} \alpha \Rightarrow [\texttt{if}\ b\ \texttt{then}\ p_1\ \texttt{else}\ p_2\ \texttt{fi}]\beta.$$

5. $p$ je `while` $b$ `do` $q$ `od`. Jako indukční předpoklad předpokládejme existenci úspěšného tabla pro formuli $\alpha_q \Rightarrow [q]\beta_q$, kde $\alpha_q$ a $\beta_q$ jsou libovolné formule bez modalit takové, že platí $\models_{\mathscr{M}} \alpha_q \Rightarrow [q]\beta_q$.

Využijeme toho, že předpokládáme existenci formule $\mathsf{wlp}(\beta \mid$ `while` $b$ `do` $q$ `od`$)$. Označme ji jako $\varphi$.

Protože platí $\models_{\mathscr{M}} \alpha \Rightarrow [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta$, je podle lemmatu 6.4.4

$$\models_{\mathscr{M}} \alpha \Rightarrow \varphi. \tag{6.19}$$

Ukážeme dále, že platí

$$\models_{\mathscr{M}} (\varphi \wedge \neg b) \Rightarrow \beta. \tag{6.20}$$

Vztahy (6.19) a (6.20) nám totiž dovolí použít pravidlo (oslabení) a začít tak úspěšné tablo:

$$\frac{\alpha \Rightarrow [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta}{\varphi \Rightarrow [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}](\varphi \wedge \neg b)}\ \text{(oslabení)}$$

Důkaz platnosti vztahu (6.20) je jednoduchý: předpokládejme, že $M \Vdash_{\mathscr{M}} (\varphi \wedge \neg b)$. Potom z definice nejslabší liberální prepodmínky a z toho, že speciálně platí $M \Vdash_{\mathscr{M}} \varphi$ dostáváme, že

$$M \Vdash_{\mathscr{M}} [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta.$$

Protože $M \Vdash_{\mathscr{M}} \neg b$, platí $M \Vdash_{\mathscr{M}} \beta$ podle pravidel pro tvorbu akcí.

Pro dokončení úspěšného tabla je zapotřebí nyní ukázat, že platí

$$\models_{\mathscr{M}} (\varphi \wedge b) \Rightarrow [q]\varphi \tag{6.21}$$

Na dosavadní tablo pak totiž navážeme pravidlem (while) a využijeme indukční předpoklad.

Důkaz vztahu (6.21): předpokládejme, že $M \Vdash_{\mathscr{M}} (\varphi \wedge b)$ a zvolme libovolnou akci $q$ z $M$ do $M'$. Z definice nejslabší liberální prepodmínky namísto vztahu $M' \Vdash_{\mathscr{M}} \varphi$, stačí ukázat, že platí

$$M' \Vdash_{\mathscr{M}} [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta. \tag{6.22}$$

Vezměme tedy libovolnou akci $\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}$ z $M'$ do $M_1$. Protože víme, že platí $M \Vdash_{\mathscr{M}} b$, existuje akce $\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}$ z $M$ do $M_1$. Protože $M \Vdash_{\mathscr{M}} \varphi$, platí z definice nejslabší liberální prepodmínky $M \Vdash_{\mathscr{M}} [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta$, a tudíž platí vztah (6.22).

Začátek úspěšného tabla pro $\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}$ tedy je:

$$\frac{\dfrac{\alpha \Rightarrow [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}]\beta}{\varphi \Rightarrow [\texttt{while}\ b\ \texttt{do}\ q\ \texttt{od}](\varphi \wedge \neg b)}\ \text{(oslabení)}}{(\varphi \wedge b) \Rightarrow [q]\varphi}\ \text{(while)}$$

■

Nejslabší liberální prepodmínku lze nalézt i pro cyklus `while`, za předpokladu, že tento cyklus není příliš složitý.

## 6.5 Cvičení

**6.5.1 Cvičení** Podejte přesnou definici pojmů *hodnota aritmetického výrazu $e$ v paměti $M$* a *hodnota Booleovského výrazu $b$ v paměti $M$*.

Pozor: identifikátory, použité ve výrazech nemusí být v paměti $M$ vůbec definovány! Uvažujte například o Booleovském výrazu $x < 1$ v paměti $M$, kde $(x, \perp_{\mathbb{Z}}) \in M$.

**6.5.2 Cvičení** Ukažte, že akční struktura pro `while` programy je *deterministická*, tj., že má pro každou akci $a$ následující vlastnost:

Jestliže $M \xrightarrow{a} M_1$ a současně $M \xrightarrow{a} M_2$, pak $M_1 = M_2$.

Ukažte, že jde o případ zobecněné Churchovy-Rosserovy vlastnosti ve smyslu definice 5.2.2. Uveďte ty formule modální logiky, které budou v zamýšleném modelu $\mathscr{M}$ zaručeně pravdivé.

**6.5.3 Cvičení** Podejte přesnou definici valuace atomických formulí v zamýšleném modelu jazyka `while` programů.

**6.5.4 Cvičení** Ukažte, že pro libovolnou modální formuli $\varphi$ platí:

1. Jestliže $\models_{\mathscr{M}} \varphi_1 \Rightarrow [p]\varphi_2$ a $\models_{\mathscr{M}} \varphi_2 \Rightarrow [q]\varphi_3$, potom $\models_{\mathscr{M}} \varphi_1 \Rightarrow [p\,;\,q]\varphi_2$.
2. Jestliže $\models_{\mathscr{M}} \varphi \Rightarrow [p]\varphi$, potom $\models_{\mathscr{M}} \varphi \Rightarrow [\texttt{while}\ b\ \texttt{do}\ p\ \texttt{od}]\varphi$.

**6.5.5 Cvičení** Ukažte , že platí:

1. $\emptyset \xrightarrow{x := 4\,;\, x := x - 1} \{(x, 3)\}$ .

   (Upozorněme, že podle konvence 6.2.1 $\emptyset$ značí paměť, kde všechny identifikátory mají hodnotu $\perp_{\mathbb{Z}}$.)

2. $\{(x, 1), (y, 4)\} \xrightarrow{\texttt{if}\ x < y\ \texttt{then}\ y := y - x\ \texttt{else}\ y := y + x\ \texttt{fi}} \{(x, 1), (y, 3)\}$ .

**6.5.6 Cvičení** Označte jako $p$ program $\texttt{while}\ x > 0\ \texttt{do}\ x := x + 1\ \texttt{od}$. Ukažte, že pro paměť $\{(x, 1)\}$ neexistuje paměť $M'$ taková, že $\{(x, 1)\} \xrightarrow{p} M'$.

**6.5.7 Cvičení** Promyslete, proč při důkazu platnosti $\models_{\mathscr{M}} ((n \geq 0) \wedge (x \geq 0)) \Rightarrow [p]y = x^n$ z příkladu 6.3.2 nevystačíme s přímočarým důkazem indukcí podle $n$.

**6.5.8 Cvičení** Označte $k := n\,;\, y := 1\,;\, \texttt{while}\ k > 0\ \texttt{do}\ y := y * x\,;\, k := k - 1\ \texttt{od}$ jako program $p$. Ukažte, že platí

$$\models_{\mathscr{M}} (n \geq 0 \wedge x \geq 0) \Rightarrow [p]y = x^n.$$

**6.5.9 Cvičení** Označte $a := 0\,;\, b := x\,;\, \texttt{while}\ b \geq y\ \texttt{do}\ b := b - y\,;\, a := a + 1\ \texttt{od}$ jako program $p$. Ukažte, že platí

$$\models_{\mathscr{M}} (x \geq 0 \wedge y \geq 0) \Rightarrow [p](x = a * y + b \wedge 0 \leq b < y).$$

**6.5.10 Cvičení** Označte $\texttt{while}\ x \geq 0\ \texttt{do}\ y := y + 1\,;\, x := x - 1\ \texttt{od}$ jako program $p$. Ukažte, že platí

$$\models_{\mathscr{M}} (x = n \wedge y = m) \Rightarrow \langle p\rangle y = m + n.$$

**6.5.11 Cvičení** Označte $\texttt{while}\ x \geq 1\ \texttt{do}\ \texttt{if}\ \texttt{even}(x)\ \texttt{then}\ x := x/2\ \texttt{else}\ x := 3 * x + 1\ \texttt{fi}\ \texttt{od}$ jako program $p$, kde $\texttt{even}(x)$ je test na sudost. Ukažte, že platí

$$\models_{\mathscr{M}} (x = n \wedge x \geq 1) \Rightarrow [p]x = 1.$$

Pozor! Platnost $\models_{\mathscr{M}} (x = n \wedge x \geq 1) \Rightarrow \langle p\rangle x = 1$ je nevyřešený problém. Vyzkoušejte si „běh“ programu $p$ pro některé hodnoty identifikátoru $x$.

Bylo ověřeno, že program $p$ se zastaví pro všechny počáteční hodnoty identifikátoru $x$ menší nebo rovny číslu $5,6 \cdot 10^{13}$. (Zdroj: `http://www.cs.unb.ca/~alopez-o/math-faq/node61.html`.)

**6.5.12 Cvičení** Popište jednotlivá pravidla použitá při konstrukci tabla z příkladu 6.4.1 a ukažte, že jde skutečně o úspěšné tablo.

**6.5.13 Cvičení** Ukažte, že vlastnosti 1. a 2. z lemmatu 6.4.4 nejslabší liberální prepodmínku charakterisují v následujícím smyslu:

> Ať $p$ je program a $\psi$ je formule bez modalit. Předpokládejme, že formule $\alpha$ bez modalit má následující dvě vlastnosti:
>
> 1. $\models_{\mathscr{M}} \alpha \Rightarrow [p]\psi$.
> 2. Pro libovolnou formuli $\varphi$ bez modalit platí: jestliže $\models_{\mathscr{M}} \varphi \Rightarrow [p]\psi$, pak $\models_{\mathscr{M}} \varphi \Rightarrow \alpha$.
>
> Potom $\models_{\mathscr{M}} \alpha \Leftrightarrow \mathsf{wlp}(\psi \mid p)$.

**6.5.14 Cvičení** Dokažte tvrzení 6.4.5.

# Kapitola 7

# Modální logika jako algebra

V této kapitole předvedeme jiný pohled na formule modální výrokové logiky. Půjde o zobecnění pohledu na klasickou výrokovou logiku jako na studium Booleových algeber. Protože modální výroková logika je složitější než výroková logika, budou výsledné algebraické struktury složitější než Booleovy algebry. Těmto novým algebraickým strukturám budeme říkat *modální algebry*. Jejich vztah k Booleovým algebrám je následující:

> Víme, že v případě $\Delta = \emptyset$ Hennessyho-Milnerova modální logika definovaná v kapitole 4 kolabuje v obyčejnou výrokovou logiku. To lze chápat i obráceně: modální logika je vlastně výroková logika obohacená o modality.
>
> Analogicky: modální algebry jsou Booleovy algebry a „ještě něco navíc“: tato dodatečná struktura nám umožní zacházet s modalitami box a diamant.

Zavedení modálních algeber nám dovolí splatit v této kapitole dluh z kapitoly 5: dokončit důkaz věty o úplnosti systému minimální modální logiky **K**.

Důkazy vět o úplnosti, tedy celý odstavec 7.6, lze při čtení textu vynechat. V kapitole 8 budeme ovšem potřebovat pojem modálních algeber, vyložený v odstavci 7.3.

Začneme zopakováním základních pojmů z teorie částečně uspořádaných množin.

## 7.1 Částečně uspořádané množiny

V dalším budeme potřebovat několik pojmů z teorie částečně uspořádaných množin. Připomeňme, že *částečně uspořádaná množina* (budeme používat běžnější výraz *poset* z aglického partially ordered set) je dvojice $\langle X, \sqsubseteq \rangle$, kde $X$ je množina a $\sqsubseteq$ je binární relace částečného uspořádání na $X$, tj. je to relace, která splňuje následující tři požadavky:

1. $\sqsubseteq$ je *reflexivní*, tj. $x \sqsubseteq x$ platí pro každé $x \in X$.
2. $\sqsubseteq$ je *transitivní*, tj. jestliže $x \sqsubseteq y$ a $y \sqsubseteq z$, potom $x \sqsubseteq z$ (pro libovolná $x, y, z \in X$).
3. $\sqsubseteq$ je *antisymetrická*, tj. jestliže $x \sqsubseteq y$ a $y \sqsubseteq x$, potom $x = y$ (pro libovolná $x, y \in X$).

Poset je tedy velmi speciální případ (orientovaného) grafu. Jsou-li dva orientované grafy $\langle X_1, \sqsubseteq_1 \rangle$ a $\langle X_2, \sqsubseteq_2 \rangle$ posety, pak homomorfismu grafů $f : \langle X_1, \sqsubseteq_1 \rangle \to \langle X_2, \sqsubseteq_2 \rangle$ říkáme *monotónní zobrazení*. Připomeňme, že monotonie $f$ tedy znamená platnost tvrzení:

> jestliže $x \sqsubseteq_1 y$, potom $f(x) \sqsubseteq_2 f(y)$ (pro libovolná $x, y \in X_1$).

Ať $\langle X, \sqsubseteq \rangle$ je poset a $M$ je podmnožina $X$. Řekneme, že

1. Prvek $y \in X$ je *dolní odhad* $M$ (v $\langle X, \sqsubseteq \rangle$), pokud platí $y \sqsubseteq m$ pro všechna $m \in M$.
2. Prvek $x \in X$ je *největší dolní odhad* $M$ (v $\langle X, \sqsubseteq \rangle$) (budeme používat častější termín *infimum* $M$), značíme $x = \inf M$, pokud platí, že $x$ je dolní odhad $M$ a navíc $x$ je největší prvek mezi všemi dolními odhady, tj. je-li dáno $y$, které je dolním odhadem $M$, pak $y \sqsubseteq x$.

Duálním pojmem je nejmenší horní odhad, neboli supremum:

1. Prvek $y \in X$ je *horní odhad* $M$ (v $\langle X, \sqsubseteq \rangle$), pokud platí $m \sqsubseteq y$ pro všechna $m \in M$.
2. Prvek $x \in X$ je *nejmenší horní odhad* $M$ (v $\langle X, \sqsubseteq \rangle$) (budeme používat častější termín *supremum* $M$), značíme $x = \sup M$, pokud platí, že $x$ je horní odhad $M$ a navíc $x$ je nejmenší prvek mezi všemi horními odhady, tj. je-li dáno $y$, které je horním odhadem $M$, pak $x \sqsubseteq y$.

Suprema ani infima nemusí existovat, viz cvičení 7.7.1. Pokud však například supremum nějaké množiny existuje, pak je určeno jednoznačně.

Poset, kde každá konečná neprázdná množina má supremum i infimum se nazývá *svaz*. Pokud má v nějakém svazu i prázdná množina supremum i infimum, nazýváme tento svaz *svazem s největším a nejmenším prvkem*. Pokud má ve svazu vůbec každá množina supremum i infimum, říkáme mu *úplný svaz*.

Připomeňme, že na svaz se lze dívat i jako na algebraickou strukturu. Řekneme, že trojice $\langle X, \sqcap, \sqcup \rangle$ tvoří *abstraktní svaz*, pokud $X$ je množina a $\sqcap : X \times X \to X$ a $\sqcup : X \times X \to X$ jsou binární operace na $X$, splňující následující axiomy:

1. *Idempotence*: platí $x \sqcap x = x$ a $x \sqcup x = x$ pro každé $x \in X$.
2. *Komutativita*: platí $x \sqcap y = y \sqcap x$ a $x \sqcup y = y \sqcup x$ pro libovolná $x, y \in X$.
3. *Asociativita*: platí $x \sqcap (y \sqcap z) = (x \sqcap y) \sqcap z$ a $x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z$ pro libovolná $x, y, z \in X$.
4. *Absorpce*: platí $x \sqcap (x \sqcup y) = x$ a $x \sqcup (x \sqcap y) = x$ pro libovolná $x, y \in X$.

Přesný vztah mezi abstraktními svazy a svazy vysvětluje cvičení 7.7.3. Budeme tedy nadále adjektivum "abstraktní" vynechávat a říkat pouze svaz.

Řekneme, že svaz $\langle X, \sqcap, \sqcup \rangle$ je *distributivní*, pokud platí:

$$x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z) \quad \text{a} \quad x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z) \quad \text{pro všechna } x, y, z \in X.$$

Řekneme, že svaz $\langle X, \sqcap, \sqcup \rangle$ s největším prvkem $\top$ a nejmenším prvkem $\bot$ je *komplementární*, pokud na $X$ existuje unární operace $^{-1} : X \to X$ zvaná *komplement*, která splňuje následující axiomy:

$$x \sqcap x^{-1} = \bot \quad \text{a} \quad x \sqcup x^{-1} = \top \quad \text{pro všechna } x \in X.$$

Prvku $x^{-1}$ říkáme *komplement prvku* $x$ (nebo také *doplněk prvku* $x$).

*Booleova algebra* je distributivní komplementární svaz s největším a největším prvkem.

## 7.2 Výroková logika jako algebra

V tomto oddíle připomeneme souvislost formulí výrokové logiky s Booleovými algebrami.

Podle cvičení 4.4.7 můžeme ztotožnit formule klasické výrokové logiky s formulemi modální výrokové logiky pro prázdný typ $\Delta = \emptyset$.

Předpokládejme, že máme dány formule výrokové logiky $Log_\emptyset(At)$ a nějaký modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$. Samozřejmě, díky tomu, že $\Delta = \emptyset$, můžeme akční strukturu $\mathbb{V}$ ztotožnit s její množinou vrcholů.

Každá formule $\varphi$ pak definuje množinu těch stavů akční struktury $\mathbb{V}$, ve kterých je $\varphi$ $\mathscr{M}$-vynucena. Danou formuli výrokové logiky pak můžeme (pouze vzhledem k modelu $\mathscr{M}$, samozřejmě) s touto množinou ztotožnit.

**7.2.1 Definice** *Extense formule $\varphi$ v modelu $\mathscr{M}$* je množina $\| \varphi \|_\mathscr{M}$ definovaná takto:

$$\| \varphi \|_\mathscr{M} = \{v \in V \mid v \Vdash_\mathscr{M} \varphi\}.$$

**7.2.2 Lemma** *Platí:*

1. $\| \mathtt{tt} \|_\mathscr{M} = V$, $\| \mathtt{ff} \|_\mathscr{M} = \emptyset$.
2. $\| a \|_\mathscr{M} = \mathscr{V}(a)$ *pro každou atomickou formuli* $a \in At$.
3. $\| \neg\varphi \|_\mathscr{M} = V \setminus \| \varphi \|_\mathscr{M}$.

4. $\| \varphi_1 \wedge \varphi_2 \|_{\mathscr{M}} = \| \varphi_1 \|_{\mathscr{M}} \cap \| \varphi_2 \|_{\mathscr{M}}$.
5. $\| \varphi_1 \|_{\mathscr{M}} \subseteq \| \varphi_2 \|_{\mathscr{M}}$ *iff* $\models_{\mathscr{M}} \varphi_1 \Rightarrow \varphi_2$.

Důkaz. Vše plyne triviálně z definice $\mathscr{M}$-vynucení. ■

Předchozí lemma nám především skutečně umožňuje ztotožnit formuli s její extensí, neboť jeho důsledkem je, že $\| \varphi_1 \|_{\mathscr{M}} = \| \varphi_2 \|_{\mathscr{M}}$ platí právě tehdy, když $\models_{\mathscr{M}} \varphi_1 \Leftrightarrow \varphi_2$. Protože formule „je“ nyní množina stavů, můžeme se ptát, jakou úlohu zde hrají logické spojky. Připomeňme, že $exp(V)$ značí množinu všech podmnožin množiny $V$.

Na model $\mathscr{M}$ můžeme pohlížet jako na množinu $exp(V)$, vybavenou následující algebraickou strukturou:

1. *Konstantami* $\| \mathtt{tt} \|_{\mathscr{M}}$, $\| \mathtt{ff} \|_{\mathscr{M}}$ a $\| a \|_{\mathscr{M}}$ pro každé $a \in At$.
2. *Unární operací* $\| \neg \|_{\mathscr{M}} : exp(V) \to exp(V)$, která každému prvku $M \in exp(V)$ (tj. každé podmnožině $M \subseteq V$) přiřadí množinu stavů $\| \neg \|_{\mathscr{M}}(M) = V \setminus M$.
3. *Binární operací* $\| \wedge \|_{\mathscr{M}} : exp(V) \times exp(V) \to exp(V)$, která každému prvku $(M, N) \in exp(V) \times exp(V)$ (tj. každé dvojici podmnožin $M \subseteq V$, $N \subseteq V$) přiřadí množinu stavů $\| \wedge \|_{\mathscr{M}}(M, N) = M \cap M$.

Výše uvedené značení nám nyní dovoluje psát např. $\| \varphi_1 \wedge \varphi_2 \|_{\mathscr{M}} = \| \wedge \|_{\mathscr{M}}(\| \varphi_1 \|_{\mathscr{M}}, \| \varphi_2 \|_{\mathscr{M}})$.

I na odvozené spojky $\vee$, $\Rightarrow$ a $\Leftrightarrow$ se lze dívat jako na binární operace na $exp(V)$. Předvedeme to pouze pro $\| \vee \|_{\mathscr{M}}$, pro ostatní spojky odkazujeme na cvičení 7.7.7. Protože $\varphi_1 \vee \varphi_2$ je syntaktická zkratka za $\neg(\neg\varphi_1 \wedge \neg\varphi_2)$, je $\| \vee \|_{\mathscr{M}}(M, N) = M \cup N$ podle definic $\| \wedge \|_{\mathscr{M}}$, $\| \neg \|_{\mathscr{M}}$ a de Morganových pravidel.

Algebraická struktura, kterou obržíme, je Booleova algebra.

**7.2.3 Lemma** *Platí:*

1. *Trojice* $\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}} \rangle$ *je distributivní svaz s největším prvkem* $\| \mathtt{tt} \|_{\mathscr{M}}$ *a nejmenším prvkem* $\| \mathtt{ff} \|_{\mathscr{M}}$.
2. $\| \neg \|_{\mathscr{M}}$ *je operace komplementu na* $\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}} \rangle$.
3. $\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}}, \| \neg \|_{\mathscr{M}}, \| \mathtt{tt} \|_{\mathscr{M}}, \| \mathtt{ff} \|_{\mathscr{M}} \rangle$ *je Booleova algebra.*

Důkaz. Zřejmé. ■

Všimněme si ale, že ne nutně každá podmnožina $V$ je extensí nějaké formule $\varphi$, jak ukazuje následující příklad.

**7.2.4 Příklad** Ať $V = \{v, w\}$ a $\mathscr{V}(a) = \emptyset$ pro všechna $a \in At$. Ukážeme, že je-li $\varphi$ formule a $M = \| \varphi \|_{\mathscr{M}}$, pak buď $M = \emptyset$ nebo $M = V$. Budeme postupovat indukcí podle složitosti formule $\varphi$.

1. Je-li $\varphi$ buď $\mathtt{tt}$ nebo $\mathtt{ff}$ nebo $a$, pak je tvrzení zřejmé.
2. Předpokládejme, že pro všechny formule $\psi$ nižší složitosti než $\varphi$ tvrzení platí. Mohou nastat dva případy:
   (a) $\varphi$ je tvaru $\neg\psi$. Pak $\| \varphi \|_{\mathscr{M}} = V \setminus \| \psi \|_{\mathscr{M}}$. Tudíž $\| \varphi \|_{\mathscr{M}} = \emptyset$ nebo $\| \varphi \|_{\mathscr{M}} = V$.
   (b) $\varphi$ je tvaru $\psi_1 \wedge \psi_2$. Pak $\| \varphi \|_{\mathscr{M}} = \| \psi_1 \|_{\mathscr{M}} \cap \| \psi_2 \|_{\mathscr{M}}$. Opět snadno plyne, že $\| \varphi \|_{\mathscr{M}} = \emptyset$ nebo $\| \varphi \|_{\mathscr{M}} = V$.

Celá Booleova algebra na množině $exp(V)$ tedy nutně nevystihuje původní modální model. Rozbor předchozího příkladu nabízí následující myšlenku: omezme se pouze na Booleovu algebru $\langle B, \cap, \cup, \setminus, V, \emptyset \rangle$, ve které je pouze to, co je nezbytně nutné, tj.: $B$ je nejmenší podmnožina $exp(V)$ taková, že

1. Množiny $V$, $\emptyset$ a $\mathscr{V}(a)$ pro každé $a \in At$ jsou prvky $B$.
2. Je-li $M \in B$, pak $V \setminus M \in B$.
3. Je-li $M \in B$, $N \in B$, pak $M \cap N \in B$.

Taková množina jistě existuje a výsledná struktura je zřejmě Booleova algebra.

**7.2.5 Definice** Ať $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot \rangle$ je Booleova algebra, $G \subseteq B$. *Booleova algebra generovaná množinou* $G$ je Booleova algebra $\mathbb{B}' = \langle B', \sqcap', \sqcup', ^{-1'}, \top', \bot' \rangle$ taková, že

1. $\mathbb{B}'$ je Booleova podalgebra $\mathbb{B}$ (neboli $\sqcap'$ je restrikce $\sqcap$, $\sqcup'$ je restrikce $\sqcup$ atd.).
2. $B'$ je nejmenší množina taková, že
   (a) $\top \in B'$, $\bot \in B'$ a $g \in B'$ pro všechna $g \in G$.
   (b) Jsou-li $b_1$ a $b_2$ v $B'$, pak i $b_1 \sqcap b_2$, $b_1 \sqcup b_2$ a $b_1^{-1}$ jsou v $B'$.

**7.2.6 Věta** *Ať $\Delta = \emptyset$, ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$ je modální model $Log_\Delta(At)$. Položme $G = \{\mathscr{V}(a) \mid a \in At\}$. Pak množina těch podmnožin $M$, které jsou extensí nějaké formule $\varphi$ v modelu $\mathscr{M}$ tvoří nosnou množinu Booleovy algebry generované množinou $G \subseteq exp(V)$.*

Důkaz. Triviální. ■

## 7.3 Modální algebry

Úvahy z předchozí části jsme sice prováděli pro speciální případ $\Delta = \emptyset$, nicméně definice extense na této speciální volbě akční signatury nijak nezávisela.

Rádi bychom notaci extense rozšířili i na poslední syntaktickou konstrukci, $[R]\varphi$, tzn. pro každé $R \in \Delta$ chceme definovat unární operaci $\| [R] \| : exp(V) \to exp(V)$ tak, aby platilo:

$$\| [R]\varphi \|_{\mathscr{M}} = \| [R] \|_{\mathscr{M}}(\| \varphi \|_{\mathscr{M}}) \quad \text{pro každé } R \in \Delta \text{ a } \varphi \in Log_\Delta(At). \tag{7.1}$$

Je jediná možnost, jak to udělat, a to:

$$\| [R] \|_{\mathscr{M}}(M) = \{v \in V \mid \text{je-li } v \xrightarrow{R} v', \text{ pak } v' \in M\}$$

Pokud shrneme výsledky dosud dosažené, zjišťujeme, že se můžeme dívat na model $\mathscr{M}$ jako na Booleovu algebru

$$\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}}, \| \neg \|_{\mathscr{M}}, \| \mathtt{tt} \|_{\mathscr{M}}, \| \mathtt{ff} \|_{\mathscr{M}} \rangle$$

vybavenou pro každé $R \in \Delta$ unární operací $\| [R] \|_{\mathscr{M}}$. Jaké jsou vlastnosti těchto unárních operací? (Viz také cvičení 7.7.8.)

**7.3.1 Tvrzení** *Pro libovolnou operaci $\| [R] \|_{\mathscr{M}}$ platí:*

1. *$\| [R] \|_{\mathscr{M}}(V) = V$, tj. operace $\| [R] \|_{\mathscr{M}}(M)$ zachovává největší prvek.*
2. *$\| [R] \|_{\mathscr{M}}(M \cap N) = \| [R] \|_{\mathscr{M}}(M) \cap \| [R] \|_{\mathscr{M}}(N)$, pro libovolné $M \subseteq V$, $N \subseteq V$, tj. operace $\| [R] \|_{\mathscr{M}}(M)$ zachovává konečné průniky.*

Důkaz. 1. Inkluse $\| [R] \|_{\mathscr{M}}(V) \subseteq V$ je zřejmá. Pro opačnou inklusi stačí ukázat, že $v \in \| [R] \|_{\mathscr{M}}(V)$ pro libovolný vrchol $v \in V$. Vezměme libovolnou akci $v \xrightarrow{R} v'$. Pak ale $v' \in V$, neboli $v \in \| [R] \|_{\mathscr{M}}(V)$ podle definice.

2. Vezměme libovolné množiny $M \subseteq V$, $N \subseteq V$.

Inkluse $\| [R] \|_{\mathscr{M}}(M \cap N) \subseteq \| [R] \|_{\mathscr{M}}(M) \cap \| [R] \|_{\mathscr{M}}(N)$. Ať $v \in \| [R] \|_{\mathscr{M}}(M \cap N)$ je libovolný vrchol. Ukážeme, že $v \in \| [R] \|_{\mathscr{M}}(M)$. Fakt, že $v \in \| [R] \|_{\mathscr{M}}(N)$ se ukáže analogicky.

Je-li $v \xrightarrow{R} v'$ libovolná akce, pak podle definice je $v' \in M \cap N$. Tedy $v' \in M$.

Inkluse $\| [R] \|_{\mathscr{M}}(M) \cap \| [R] \|_{\mathscr{M}}(N) \subseteq \| [R] \|_{\mathscr{M}}(M \cap N)$. Ať $v \in \| [R] \|_{\mathscr{M}}(M) \cap \| [R] \|_{\mathscr{M}}(N)$ je libovolný vrchol.
Je-li $v \xrightarrow{R} v'$ libovolná akce, pak $v' \in M$ a současně $v' \in N$, neboli $v \in \| [R] \|_{\mathscr{M}}(M \cap N)$.

■

Vlastnosti těchto unárních operací $\| [R] \|_{\mathscr{M}}$ z tvrzení 7.3.1 nyní axiomatizujeme a dostaneme tak pojem *modální algebry*, který má analogický vztah k modelům modální logiky, jako mají Booleovy algebry k modelům výrokové logiky.

**7.3.2 Definice** *Modální algebra typu* $\Delta$ je Booleova algebra $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot \rangle$, která je vybavena unární operací $[\![R]\!] : B \to B$ pro každé $R \in \Delta$ tak, že pro všechna $R \in \Delta$ a pro všechna $x, y \in B$ jsou splněny následující dvě rovnosti:

$$\begin{aligned} [\![R]\!](\top) &= \top \\ [\![R]\!](x \sqcap y) &= [\![R]\!](x) \sqcap [\![R]\!](y) \end{aligned}$$

Podobně jako diamant je syntaktická zkratka za jistou kombinaci negací a boxu, zavedeme na každé modální algebře *odvozenou* unární operaci $\langle\!\langle R \rangle\!\rangle$ pro každou akci $R$ následovně:

$$\langle\!\langle R \rangle\!\rangle(x) = ([\![R]\!](x^{-1}))^{-1} \quad \text{pro každé } x \in B. \tag{7.2}$$

Tato odvozená operace má zřejmě následující vlastnosti:

**7.3.3 Lemma** *Pro každou akci $R$ v modální algebře platí:*

1. $\langle\!\langle R \rangle\!\rangle(\bot) = \bot$.
2. $\langle\!\langle R \rangle\!\rangle(x \sqcup y) = \langle\!\langle R \rangle\!\rangle(x) \sqcup \langle\!\langle R \rangle\!\rangle(y)$ *pro všechna $x$, $y$.*

Důkaz. Důkaz plyne z toho, že na každé modální algebře je také struktura Booleovy algebry. ■

## 7.4 Korespondence Booleových algeber a modelů výrokové logiky

V této části ukážeme, že přechod od modálních modelů pro typ $\Delta = \emptyset$ k Booleovým algebrám lze obrátit. Pro každou Booleovu algebru tedy sestrojíme jistý modální model. Dostaneme tak korespondenci mezi modálními modely typu $\emptyset$ a Booleovými algebrami.

Následující úvahy budou zobecněním cvičení 4.4.7.

Nejprve zobecněme pojem ohodnocení formulí výrokové logiky na případ, kdy množinou „pravdivostních hodnot“ je libovolná Booleova algebra.

**7.4.1 Definice** Ať $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot \rangle$ je Booleova algebra. $\mathbb{B}$*-ohodnocení formulí výrokové logiky* je funkce $u : Log_{\emptyset}(At) \to B$ splňující následující podmínky:

1. $u(\mathtt{tt}) = \top$.
2. $u(\neg\varphi) = u(\varphi)^{-1}$.
3. $u(\varphi \wedge \psi) = u(\varphi) \sqcap u(\psi)$.

Řekneme, že formule $\varphi$ je *pravdivá v* $\mathbb{B}$*-ohodnocení* $u$, pokud platí $u(\varphi) = \top$.

Uvědomme si, že předchozí definice ohodnocení v obecné Booleově algebře je přímým zobecněním pojmu ohodnocení ve dvouprvkové Booleově algebře na množině $\{0, 1\}$. Podobně jako v klasické definici ohodnocení platí následující tvrzení, které říká, že stačí ohodnotit atomické formule.

**7.4.2 Lemma** *Ať $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot \rangle$ je Booleova algebra. Potom platí:*

*Každé zobrazení $u : At \to B$ indukuje právě jedno $\mathbb{B}$-ohodnocení $u^\sharp : Log_{\emptyset}(At) \to B$ takové, že $u(a) = u^\sharp(a)$ pro všechna $a \in At$.*

Důkaz. Důkaz rozdělíme na dvě části. Nejprve ukážeme, že takové $\mathbb{B}$-ohodnocení $u^\sharp$ existuje. Potom ukážeme, že je právě jedno.

1. Zobrazení $u^\sharp : Log_\emptyset(At) \to B$ definujeme indukcí podle syntaktické složitosti formule $\varphi$.

   - $\varphi = \mathtt{tt}$. Položme $u^\sharp(\mathtt{tt}) = \top$.
   - $\varphi = a$, $a \in At$. Položme $u^\sharp(a) = u(a)$.
   - $\varphi = \varphi_1 \wedge \varphi_2$. Položme $u^\sharp(\varphi) = u^\sharp(\varphi_1) \sqcap u^\sharp(\varphi_2)$.

   Toto zobrazení $u^\sharp$ je zřejmě $\mathbb{B}$-ohodnocení splňující $u^\sharp(a) = u(a)$ pro všechna $a \in At$.

2. Předpokládejme, že $v : Log_\emptyset(At) \to B$ je $\mathbb{B}$-ohodnocení splňující $v(a) = u(a)$ pro všechna $a \in At$. Chceme ukázat, že $v(\varphi) = u^\sharp(\varphi)$ pro libovolnou formuli $\varphi$. Důkaz opět provedeme indukcí podle syntaktické složitosti formule $\varphi$.

   - $\varphi = \mathtt{tt}$. Potom $v(\mathtt{tt}) = u^\sharp(\mathtt{tt}) = \top$, protože $v$ i $u^\sharp$ jsou $\mathbb{B}$-ohodnocení.
   - $\varphi = a$, $a \in At$. Potom rovnost $v(a) = u^\sharp(a)$ platí z předpokladů.
   - $\varphi = \varphi_1 \wedge \varphi_2$. Předpokládejme, že rovnosti $v(\varphi_1) = u^\sharp(\varphi_1)$ a že $v(\varphi_2) = u^\sharp(\varphi_2)$ platí. Platnost rovnosti $v(\varphi) = u^\sharp(\varphi)$ pak plyne z toho, že $\sqcap$ je operace na $B$.

■

Hodnoty $\mathbb{B}$-ohodnocení pro odvozené spojky jsou následující:

**7.4.3 Lemma** *Ať $u : Log_\emptyset(At) \to B$ je $\mathbb{B}$-ohodnocení v Booleově algebře $\mathbb{B} = \langle B, \sqcap, \sqcup, {}^{-1}, \top, \bot\rangle$. Potom platí:*

1. $u(\mathtt{ff}) = \bot$.
2. $u(\varphi \vee \psi) = u(\varphi) \sqcup u(\psi)$.
3. $u(\varphi \Rightarrow \psi) = u(\varphi)^{-1} \sqcup u(\psi)$.
4. $u(\varphi \Leftrightarrow \psi) = u(\varphi \Rightarrow \psi) \sqcap u(\psi \Rightarrow \varphi)$.

Důkaz. Plyne snadno z vlastností Booleových algeber. ■

**7.4.4 Lemma** *Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ je modální model logiky $Log_\emptyset(At)$. Označme jako $\mathbb{B}$ Booleovu algebru*

$$\langle exp(V), \| \wedge \|_\mathscr{M}, \| \vee \|_\mathscr{M}, \| \neg \|_\mathscr{M}, \| \mathtt{tt} \|_\mathscr{M}, \| \mathtt{ff} \|_\mathscr{M}\rangle.$$

*Valuace $\mathscr{V}$ modelu $\mathscr{M}$ pak určuje $\mathbb{V}$-ohodnocení $u$ formulí $Log_\emptyset(At)$ následovně:*

1. $u(a) = \mathscr{V}(a)$ *pro všechna* $a \in At$.
2. $u(\mathtt{tt}) = V$.
3. $u(\neg\varphi) = V \setminus u(\varphi)$.
4. $u(\varphi_1 \wedge \varphi_2) = u(\varphi_1) \cap u(\varphi_2)$.

Důkaz. Triviální. ■

Pokusíme se nyní předchozí úvahy obrátit. Začneme s Booleovou algebrou $\mathbb{B}$ a $\mathbb{B}$-ohodnocením $u$ formulí $Log_\emptyset(At)$ a zkonstruujeme modální model $\mathscr{M}_\mathbb{B} = \langle \mathbb{V}_\mathbb{B}, \mathscr{V}_u\rangle$.

**7.4.5 Konstrukce** Předpokládejme, že $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot \rangle$ je Booleova algebra a $u : Log_\emptyset(At) \to B$ je $\mathbb{B}$-ohodnocení.

Vrcholy akční struktury $\mathbb{V}_\mathbb{B}$ budou jisté systémy podmnožin množiny $B$.

Intuitivně si můžeme každý prvek množiny $B$ představit jako nějakou formuli výrokové logiky. Vrcholem $U$ akční struktury $\mathbb{V}_\mathbb{B}$ potom bude systém všech výroků „pravdivých ve vrcholu“ $U$. Předpokládejme, že takový vrchol $U$ jsme již zkonstruovali. Jak poznat, že nějaký výrok $x$ (tj. nějaký prvek $x \in B$) je pravdivý ve vrcholu $U$ (tj. $x \in U$)?

Zřejmě musí být splněny následující podmínky:

1. Platí $\top \in U$. To intuitivně odpovídá tomu, že formuli `tt` modelujeme v Booleově algebře $\mathbb{B}$ konstantou $\top$.
2. Platí $\bot \notin U$. To intuitivně odpovídá tomu, že formuli `ff` modelujeme v Booleově algebře $\mathbb{B}$ konstantou $\bot$.
3. $x \in U$ a současně $y \in U$ platí právě tehdy, když $(x \sqcap y) \in U$. To intuitivně odpovídá tomu, že spojku *konjunkce* modelujeme v Booleově algebře $\mathbb{B}$ operací $\sqcap$.
4. $x \in U$ nebo $y \in U$ platí právě tehdy, když $(x \sqcup y) \in U$. To intuitivně odpovídá tomu, že spojku *disjunkce* modelujeme v Booleově algebře $\mathbb{B}$ operací $\sqcup$.
5. $x \notin U$ platí právě tehdy, když $x^{-1} \in U$. To intuitivně odpovídá tomu, že spojku *negace* modelujeme v Booleově algebře $\mathbb{B}$ operací $^{-1}$.

Podmnožiny $U$ Booleovy algebry $\mathbb{B}$, splňující předchozí podmínky, se nazývají *ultrafiltry na* $\mathbb{B}$.
Akční struktura $\mathbb{V}_\mathbb{B}$ typu $\emptyset$ je tedy definována následovně:

1. Vrcholy $\mathbb{V}_\mathbb{B}$ jsou přesně všechny ultrafiltry na $\mathbb{B}$.
2. Akce ve $\mathbb{V}_\mathbb{B}$ žádné nejsou.

Právě definovaná akční struktura je zřejmě typu $\emptyset$. Modální model $\mathscr{M}_\mathbb{B}$ na $\mathbb{V}_\mathbb{B}$ je dán následující valuací:

$$\mathscr{V}_u(a) = \{U \mid u(a) \in U\} \quad \text{pro každé } a \in At.$$

**7.4.6 Lemma** *Ať $\mathscr{M}_\mathbb{B} = \langle \mathbb{V}_\mathbb{B}, \mathscr{V}_u \rangle$ je modální model vzniklý z Booleovy algebry $\mathbb{B}$ a ohodnocení $u$ konstrukcí 7.4.5. Potom pro libovolný vrchol $U$ a libovolnou formuli platí:*

$U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi$ *platí právě tehdy, když* $u(\varphi) \in U$.

Důkaz. Budeme postupovat indukcí podle syntaktické složitosti formule $\varphi$.

1. $\varphi = \mathtt{tt}$. Potom tvrzení plyne z toho, že $u(\mathtt{tt}) = \top$ a z toho, že pro každý ultrafiltr $U$ platí $\top \in U$.
2. $\varphi = a$, $a \in At$. $U \Vdash_{\mathscr{M}_\mathbb{B}} a$ znamená přesně to, že $U \in \mathscr{V}_u(a)$. To je ale přesně to, že $u(a) \in U$.
3. $\varphi = \neg\varphi_1$. Platnost $U \Vdash_{\mathscr{M}_\mathbb{B}} \neg\varphi_1$ znamená právě to, že $U \not\Vdash_{\mathscr{M}_\mathbb{B}} \varphi_1$, což podle indukčního předpokladu je přesně to, že $u(\varphi_1) \notin U$. Poslední vztah platí právě tehdy, když $u(\varphi_1)^{-1} \in U$, protože $U$ je ultrafiltr. Podle definice ohodnocení je ale $u(\varphi_1)^{-1} = u(\neg\varphi_1)$.
4. $\varphi = \varphi_1 \wedge \varphi_2$. Platnost $U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi_1 \wedge \varphi_2$ je přesně to, že platí $U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi_1$ a současně $U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi_2$. Podle indukčního předpokladu to je právě tehdy, když $u(\varphi_1) \in U$ a současně $u(\varphi_2) \in U$. Protože $U$ je ultrafiltr, je to právě tehdy, když $u(\varphi_1 \wedge \varphi_2) \in U$.

■

Konstrukce 7.4.5 spolu s lemmatem 7.4.4 nám dává následující výsledek:

**7.4.7 Věta** *Pro výrokovou logiku (tj. pro modální logiku, kdy $\Delta = \emptyset$) máme následující dvě konstrukce:*

1. *Každý modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$ dává vzniknout Booleově algebře $\mathbb{B}_\mathscr{M}$ a $\mathbb{B}_\mathscr{M}$-ohodnocení $u$ popsaném v lemmatu 7.4.4.*
2. *Každá Booleova algebra $\mathbb{B}$ a ohodnocení $u$ dává vzniknout modálnímu modelu $\mathscr{M}_\mathbb{B}$, popsanému v konstrukci 7.4.5.*

## 7.5 Korespondence modálních algeber a modálních modelů

V tomto odstavci dokážeme výsledky analogické výsledkům odstavce 7.4. Připomeňme, že modální algebra typu $\Delta$ je Booleova algebra vybavená sadou unárních operací $[\![R]\!]$ (definice 7.3.2).

**7.5.1 Definice** Ať $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![R]\!]\rangle$ je modální algebra typu $\Delta$. $\mathbb{B}$-*ohodnocení formulí logiky* $Log_\Delta(At)$ je funkce $u : Log_\Delta(At) \to B$ splňující následující podmínky:

1. $u(\mathtt{tt}) = \top$.
2. $u(\neg\varphi) = u(\varphi)^{-1}$.
3. $u(\varphi \wedge \psi) = u(\varphi) \sqcap u(\psi)$.
4. $u([R]\varphi) = [\![R]\!](u(\varphi))$.

Řekneme, že formule $\varphi$ je *pravdivá v* $\mathbb{B}$-*ohodnocení* $u$, pokud platí $u(\varphi) = \top$.

$\mathbb{B}$-ohodnocení odvozených spojek lze spočítat následovně:

**7.5.2 Lemma** *Ať $u$ je $\mathbb{B}$-ohodnocení, ať $\varphi$ a $\psi$ jsou formule. Pak platí:*

1. $u(\mathtt{ff}) = \bot$.
2. $u(\varphi \vee \psi) = u(\varphi) \sqcup u(\psi)$.
3. $u(\varphi \Rightarrow \psi) = u(\varphi)^{-1} \sqcup u(\psi)$.
4. $u(\varphi \Leftrightarrow \psi) = u(\varphi \Rightarrow \psi) \sqcap u(\psi \Rightarrow \varphi)$.
5. $u(\langle R\rangle(\varphi)) = \langle\!\langle R\rangle\!\rangle(u(\varphi))$.

Důkaz. Důkaz plyne snadno z definice $\mathbb{B}$-ohodnocení a toho, že odvozené spojky jsou syntaktické zkratky. ■

Analogií lemmatu 7.4.2 je následující výsledek.

**7.5.3 Lemma** *Ať $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![R]\!]\rangle$ je modální algebra typu $\Delta$. Potom platí:*

*Každé zobrazení $u : At \to B$ indukuje právě jedno $\mathbb{B}$-ohodnocení $u^\sharp : Log_\Delta(At) \to B$ takové, že $u(a) = u^\sharp(a)$ pro všechna $a \in At$.*

Důkaz. Postupujte podobně jako v důkazu lemmatu 7.4.2. Jediné, co je zde navíc, je definice $u^\sharp$ pro modalitu box. Položte $u^\sharp([R]\varphi) = [\![R]\!](u^\sharp(\varphi))$. ■

I důkaz následujícího tvrzení je triviální.

**7.5.4 Lemma** *Ať $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ je modální model logiky $Log_\Delta(At)$. Označme jako $\mathbb{B}$ modální algebru*

$$\langle exp(V), \|\wedge\|_{\mathscr{M}}, \|\vee\|_{\mathscr{M}}, \|\neg\|_{\mathscr{M}}, \|\,\mathtt{tt}\,\|_{\mathscr{M}}, \|\,\mathtt{ff}\,\|_{\mathscr{M}}, \|\,[R]\,\|_{\mathscr{M}}\rangle.$$

*Valuace $\mathscr{V}$ modelu $\mathscr{M}$ pak určuje $\mathbb{V}$-ohodnocení $u$ formulí $Log_\Delta(At)$ následovně:*

1. $u(a) = \mathscr{V}(a)$ *pro všechna* $a \in At$.
2. $u(\mathtt{tt}) = V$.
3. $u(\neg\varphi) = V \setminus u(\varphi)$.
4. $u(\varphi_1 \wedge \varphi_2) = u(\varphi_1) \cap u(\varphi_2)$.
5. $u([R]\varphi) = \|\,[R]\,\|_{\mathscr{M}}(u(\varphi))$.

**7.5.5 Konstrukce** Předpokládejme, že je dána modální algebra $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![R]\!]\rangle$ a $\mathbb{B}$-ohodnocení $u : Log_\Delta(At) \to B$.

Akční struktura $\mathbb{V}_\mathbb{B}$ typu $\Delta$ je definována následovně:

1. Vrcholy $\mathbb{V}_\mathbb{B}$ jsou přesně všechny ultrafiltry na $\mathbb{B}$. To vyplývá z toho, že formule `tt`, `ff` a spojky negace, konjunkce a disjunkce interpretujeme v modální algebře stejně, jako v Booleově algebře (viz konstrukce 7.4.5).

   Přesná definice ultrafiltru na modální algebře je tedy následující:

   Podmnožina $U \subseteq B$ je *ultrafiltr na modální algebře* $\mathbb{B}$, pokud platí:
   (a) Platí $\top \in U$.
   (b) Platí $\bot \notin U$.
   (c) $x \in U$ a současně $y \in U$ platí právě tehdy, když $(x \sqcap y) \in U$.
   (d) $x \in U$ nebo $y \in U$ platí právě tehdy, když $(x \sqcup y) \in U$.
   (e) $x \notin U$ platí právě tehdy, když $x^{-1} \in U$.

2. Akce $U \xrightarrow{R} U'$ ve $\mathbb{V}_\mathbb{B}$ je právě tehdy, když platí:

   $[\![R]\!](x) \in U$ právě tehdy, když $x \in U'$.

Právě definovaná akční struktura je zřejmě typu $\Delta$. Modální model $\mathscr{M}_\mathbb{B}$ na $\mathbb{V}_\mathbb{B}$ je dán následující valuací:

$$\mathscr{V}_u(a) = \{U \mid u(a) \in U\} \quad \text{pro každé } a \in At.$$

Vynucení v modelu $\mathscr{M}_\mathbb{B}$ lze popsat analogicky jako v lemmatu 7.4.6.

**7.5.6 Lemma** *Ať $\mathscr{M}_\mathbb{B} = \langle \mathbb{V}_\mathbb{B}, \mathscr{V}_u\rangle$ je modální model vzniklý z modální algebry $\mathbb{B}$ a ohodnocení $u$ konstrukcí 7.5.5. Potom pro libovolný vrchol $U$ a libovolnou formuli platí:*

$U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi$ *platí právě tehdy, když* $u(\varphi) \in U$.

Důkaz. Budeme postupovat indukcí podle syntaktické složitosti formule $\varphi$. Pro atomické formule, formuli `tt` a spojky negace, konjunkce lze zřejmě postupovat podobně jako v důkazu lemmatu 7.4.6.

Zbývá tedy pouze případ, kdy $\varphi = [R]\varphi_1$. Předpokládejme, že $U \Vdash_{\mathscr{M}_\mathbb{B}} [R]\varphi_1$. Vezměme libovolnou akci $U \xrightarrow{R} U'$. Potom $U' \Vdash_{\mathscr{M}_\mathbb{B}} \varphi_1$ a podle indukčního předpokladu $u(\varphi_1) \in U'$. To nastává právě tehdy, když $[\![R]\!](u(\varphi_1)) \in U$. Protože $[\![R]\!](u(\varphi_1)) = u([R]\varphi_1)$, ukázali jsme, že $u([R]\varphi_1) \in U$.

Předpokládejme obráceně, že $u([R]\varphi_1) \in U$ a vezměme libovolnou akci $U \xrightarrow{R} U'$. Protože platí $u([R]\varphi_1) = [\![R]\!](u(\varphi_1))$, je $u(\varphi_1) \in U'$ podle definice akce $R$. Podle indukčního předpokladu platí vztah $U' \Vdash_{\mathscr{M}_\mathbb{B}} \varphi_1$. Ukázali jsme, že platí $U \Vdash_{\mathscr{M}_\mathbb{B}} [R]\varphi_1$. ■

## 7.6 Věty o úplnosti

V tomto odstavci dokážeme větu 5.1.3 o úplnosti pro systém **K** z kapitoly 5. Využijeme přitom korespondenci modálních modelů a modálních algeber.

Nejprve ukážeme, že modální tautologie (tj. ty formule $\varphi$, pro které platí $\models \varphi$) jsou přesně ty formule, které jsou tautologie ve smyslu ohodnocení. To říká následující věta.

**7.6.1 Věta** *Pro libovolnou formuli $\varphi$ jsou následující podmínky ekvivalentní:*

1. $\models \varphi$
2. *Pro libovolnou modální algebru $\mathbb{B}$ a libovolné $\mathbb{B}$-ohodnocení $u$ platí* $u(\varphi) = \top$.

Důkaz. 1. $\Rightarrow$ 2. Vezměme libovolnou modální algebru $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![R]\!]\rangle$ typu $\Delta$ a libovolné $\mathbb{B}$-ohodnocení $u : Log_\Delta(At) \to B$.

K modální algebře sestrojme modální model $\mathscr{M}_\mathbb{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ tak, jako v konstrukci 7.5.5. Podle předpokladu platí $\models_{\mathscr{M}_\mathbb{B}} \varphi$, tedy platí $U \Vdash_{\mathscr{M}_\mathbb{B}} \varphi$ pro libovolný ultrafiltr $U$. Podle lemmatu 7.5.6 dostáváme, že $u(\varphi) \in U$, pro libovolný ultrafiltr $U$. Jinými slovy: ukázali jsme, že $u(\varphi) \in \bigcap \{U \mid U \text{ je ultrafiltr}\}$.

K tomu, abychom ukázali, že $u(\varphi) = \top$, stačí ukázat:

jestliže $x \in \bigcap \{U \mid U \text{ je ultrafiltr}\}$, pak $x = \top$.

Předpokládejme, že $x \neq \top$. Potom $x^{-1} \neq \bot$ a existuje ultrafiltr $V$ takový, že $x^{-1} \in V$ (viz cvičení 7.7.9). Pak platí $x \in V$ a současně $x^{-1} \in V$. Z definice ultrafiltru toto není možné, a proto $x = \top$.

2. $\Rightarrow$ 1. Tato implikace je triviální. ■

Nejprve dokážeme algebraickou větu o korektnosti pro deduktivní systém **K** (srovnejte s první částí důkazu věty 5.1.3).

**7.6.2 Věta** *Jestliže* $\mathbf{K} \vdash \varphi$, *potom* $u(\varphi) = \top$ *pro libovolnou modální algebru* $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![R]\!]\rangle$ *typu* $\Delta$ *a libovolné* $\mathbb{B}$*-ohodnocení* $u : Log_\Delta(At) \to B$.

Důkaz. Důkaz rozdělíme na dvě části.

1. Instance každého axiomu systému **K** je ohodnocena $\top$.
   (a) Zřejmě $u(\mathtt{tt}) = \top$.
   (b) $u(\varphi \Rightarrow (\psi \Rightarrow \varphi)) = u(\varphi)^{-1} \sqcup u(\psi \Rightarrow \varphi) = u(\varphi)^{-1} \sqcup (u(\psi)^{-1} \sqcup u(\varphi)) = \top$ podle pravidel pro počítání v Booleově algebře.
   (c) Fakt, že $u((\varphi \Rightarrow (\psi_1 \Rightarrow \psi_2)) \Rightarrow ((\varphi \Rightarrow \psi_1) \Rightarrow (\varphi \Rightarrow \psi_2))) = \top$, přenecháváme jako cvičení 7.7.10.
   (d) Fakt, že $u((\neg\varphi \Rightarrow \neg\psi) \Rightarrow (\psi \Rightarrow \varphi)) = \top$, přenecháváme jako cvičení 7.7.10.
   (e) Ukážeme, že $u([R](\varphi \Rightarrow \psi) \Rightarrow ([R]\varphi \Rightarrow [R]\psi)) = \top$.
   Nejprve si uvědomme, že formuli
   $$[R](\varphi \Rightarrow \psi) \Rightarrow ([R]\varphi \Rightarrow [R]\psi)$$
   lze nahradit formulí
   $$\langle R\rangle(\varphi \wedge \neg\psi) \vee (\langle R\rangle\neg\varphi \vee [R]\psi)$$
   (využili jsme definici implikace a diamantu jako syntaktických zkratek).
   V následujícím výpočtu použijeme lemma 7.5.2 a počítání v modální algebře:
   $$\begin{aligned} u([R](\varphi \Rightarrow \psi) \Rightarrow ([R]\varphi \Rightarrow [R]\psi)) &= u(\langle R\rangle(\varphi \wedge \neg\psi) \vee (\langle R\rangle\neg\varphi \vee [R]\psi)) \\ &= \langle\!\langle R\rangle\!\rangle(u(\varphi \wedge \neg\psi) \sqcup u(\neg\psi)) \sqcup [\![R]\!](u(\psi)) \\ &= ([\![R]\!](u(\psi)))^{-1} \sqcup [\![R]\!](u(\psi)) \\ &= \top. \end{aligned}$$
2. Odvozovací pravidla deduktivního systému **K** jsou korektní v následujícím smyslu: je-li každý předpoklad pravidla ohodnocen $\top$, pak je také závěr pravidla ohodnocen $\top$.
   (a) Pravidlo Modus Ponens. Předpokládejme, že platí $u(\varphi \Rightarrow \psi) = \top$ a $u(\varphi) = \top$. Potom musí platit $u(\psi) = \top$, protože $u(\varphi \Rightarrow \psi) = u(\varphi)^{-1} \sqcup u(\psi)$.
   (b) Pravidlo zavedení boxu. Předpokládejme, že platí $u(\varphi) = \top$. Potom $u([R]\varphi) = [\![R]\!](u(\varphi)) = [\![R]\!](\top) = \top$, podle pravidel počítání v modální algebře.

Důkaz věty lze nyní snadno dokončit indukcí podle délky důkazu formule $\varphi$ v deduktivním systému **K**. ■

Pro důkaz věty o úplnosti sestrojíme takzvanou *Lindenbaumovu-Tarského algebru* logiky $Log_\Delta(At)$. Jedná se vlastně o příklad *syntaktického modelu* pro deduktivní systém.

**7.6.3 Konstrukce** Připomeňme, že $\mathbf{K} \vdash \varphi$ znamená, že formule $\varphi$ je odvoditelná v deduktivním systému **K**. Na množině $Log_\Delta(At)$ zavedeme následující binární relaci:

$\varphi \sim \psi$ platí právě tehdy, když $\mathbf{K} \vdash \varphi \Leftrightarrow \psi$.

Je snadné ukázat, že $\sim$ je relace ekvivalence na množině $Log_\Delta(At)$ (viz cvičení 7.7.11).

Podle lemmatu 2.1.10 určuje relace $\sim$ rozklad množiny $Log_\Delta(At)$ na třídy ekvivalence. Označme $[\varphi]_\sim$ třídu ekvivalence formule $\varphi$.

Modální algebru $\mathbb{K} = \langle K, \sqcap, \sqcup, {}^{-1}, \top, \bot, [\![R]\!]\rangle$ typu $\Delta$ budeme definovat následovně:

1. Prvky množiny $K$ jsou třídy ekvivalence $[\varphi]_\sim$.
2. Konstanty $\top$ a $\bot$ definujeme jako $[\mathtt{tt}]_\sim$ a $[\mathtt{ff}]_\sim$.
3. Operace ${}^{-1}$, $\sqcap$ a $\sqcup$ jsou definovány pomocí spojek negace, konjunkce a disjunkce:
$$[\varphi]_\sim{}^{-1} = [\neg\varphi]_\sim \quad \text{a} \quad [\varphi_1]_\sim \sqcap [\varphi_2]_\sim = [\varphi_1 \wedge \varphi_2]_\sim \quad \text{a} \quad [\varphi_1]_\sim \sqcup [\varphi_2]_\sim = [\varphi_1 \vee \varphi_2]_\sim.$$
4. Pro každou akci $R$ je operace $[\![R]\!]$ definována pomocí spojky box:
$$[\![R]\!]([\varphi]_\sim) = [[R]\varphi]_\sim.$$

Je snadné ukázat, že výše uvedené definice jsou korektní a že jsme takto definovali modální algebru typu $\Delta$ (viz cvičení 7.7.11).

Definujme funkci $u : Log_\Delta(At) \to K$ následovně: $u(\varphi) = [\varphi]_\sim$. Tímto způsobem jsme zřejmě definovali $\mathbb{K}$-ohodnocení.

Povšimněme si, že podle definice platí $u(\varphi) = \top$ právě tehdy, když $\mathbf{K} \vdash \varphi \Leftrightarrow \mathtt{tt}$. Podle cvičení 5.4.2 však $\mathbf{K} \vdash \varphi \Leftrightarrow \mathtt{tt}$ platí právě tehdy, když platí $\mathbf{K} \vdash \varphi$. Hodnotou $\top$ jsou tedy ohodnoceny přesně ty formule $\varphi$, pro které platí $\mathbf{K} \vdash \varphi$.

**7.6.4 Věta** *Jestliže $u(\varphi) = \top$ pro libovolnou modální algebru $\mathbb{B} = \langle B, \sqcap, \sqcup, {}^{-1}, \top, \bot, [\![R]\!]\rangle$ typu $\Delta$ a libovolné $\mathbb{B}$-ohodnocení $u : Log_\Delta(At) \to B$, potom $\mathbf{K} \vdash \varphi$.*

Důkaz. Vezměme formuli $\varphi$ a předpokládejme, že $u(\varphi) = \top$ pro libovolnou modální algebru $\mathbb{B} = \langle B, \sqcap, \sqcup, {}^{-1}, \top, \bot, [\![R]\!]\rangle$ typu $\Delta$ a libovolné $\mathbb{B}$-ohodnocení $u : Log_\Delta(At) \to B$.

Potom $u(\varphi) = \top$ pro ohodnocení formule $\varphi$ v modální algebře $\mathbb{K}$ sestrojené v konstrukci 7.6.3. To ale znamená, že $\mathbf{K} \vdash \varphi$. ■

Zatím jsme ukázali, že dokazatelnost formule v systému **K** a její pravdivost ve smyslu ohodnocení v modálních algebrách znamená totéž.

Věty 7.6.4 a 7.6.1 nám umožňují dokončit důkaz věty 5.1.3:

**7.6.5 Věta (Věta o úplnosti pro systém K.)** *Předpokládejme, že $\models \varphi$. Potom $\mathbf{K} \vdash \varphi$.*

## 7.7 Cvičení

**7.7.1 Cvičení** Ukažte, že množina všech přirozených čísel $\mathbb{N}$ s obvyklou relací uspořádání $\leq$ tvoří poset.

1. V tomto posetu například množina všech sudých čísel nemá supremum.
2. Charakterisujte ty podmnožiny $\mathbb{N}$, které mají supremum.
3. Charakterisujte ty podmnožiny $\mathbb{N}$, které mají infimum.

**7.7.2 Cvičení** Existuje poset, kde každá dvojice různých prvků má supremum, ale žádná dvojice různých prvků nemá infimum?

**7.7.3 Cvičení** V tomto cvičení se ukáže, že svazy a abstraktní svazy si vzájemně jednoznačně korespondují.

1. Je-li poset $\langle X, \sqsubseteq\rangle$ svaz, pak lze definovat binární operace na $X$ předpisy $\sqcap(x,y) = \inf\{x,y\}$, $\sqcup(x,y) = \sup\{x,y\}$ a trojice $\langle X, \sqcap, \sqcup\rangle$ je abstraktní svaz. Označme tento abstraktní svaz jako $L\langle X, \sqsubseteq\rangle$.

2. Ať trojice $\langle X, \sqcap, \sqcup\rangle$ je abstraktní svaz. Ukažte, že platí $x \sqcap y = x$ iff $x \sqcup y = y$ (pro libovolná $x, y \in X$). Definujte binární relaci $\sqsubseteq$ na množině $X$ takto: $x \sqsubseteq y$ iff $x \sqcap y = x$. Ukažte, že dvojice $\langle X, \sqsubseteq\rangle$ je poset, který je svaz. Označme tento svaz jako $P\langle X, \sqcap, \sqcup\rangle$.

3. Ukažte, že přiřazení
$$\langle X, \sqsubseteq\rangle \mapsto L\langle X, \sqsubseteq\rangle \quad \text{a} \quad \langle X, \sqcap, \sqcup\rangle \mapsto P\langle X, \sqcap, \sqcup\rangle$$
jsou vzájemně inversní, tj. platí: $PL\langle X, \sqsubseteq\rangle = \langle X, \sqsubseteq\rangle$ a $LP\langle X, \sqcap, \sqcup\rangle = \langle X, \sqcap, \sqcup\rangle$.

**7.7.4 Cvičení** Ukažte, že pro poset $\langle X, \sqsubseteq\rangle$ jsou následující podmínky ekvivalentní:

1. Každá podmnožina $M \subseteq X$ má supremum.

2. Každá podmnožina $M \subseteq X$ má infimum.

Návod pro implikaci 1. $\Rightarrow$ 2.: pro konstrukci infima množiny $M$ sestrojte nejprve množinu $M^{\nabla}$ všech dolních odhadů množiny $M$. Podle předpokladu existuje $\sup M^{\nabla}$, označte jej $x$. Ukažte, že $x = \inf M$.

Implikaci 2. $\Rightarrow$ 1. dokažte analogicky.

**7.7.5 Cvičení** Ukažte, že ve svazu $\langle X, \sqcap, \sqcup\rangle$ stačí předpokládat pouze jeden z distributivních zákonů, tj. ukažte, že v každém svazu jsou následující podmínky ekvivalentní:

1. Rovnost $x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$ platí pro všechna $x, y, z \in X$.

2. Rovnost $x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z)$ platí pro všechna $x, y, z \in X$.

**7.7.6 Cvičení** Dokažte, že $\| \varphi_1 \|_{\mathscr{M}} = \| \varphi_2 \|_{\mathscr{M}}$ platí právě tehdy, když $\models_{\mathscr{M}} \varphi_1 \Leftrightarrow \varphi_2$.

**7.7.7 Cvičení** Definujte $\| \Rightarrow \|_{\mathscr{M}}$ a $\| \Leftrightarrow \|_{\mathscr{M}}$ jako binární operace na $exp(V)$. Dále definujte $\| \langle R\rangle \|_{\mathscr{M}}$ jako unární operaci na $exp(V)$.

**7.7.8 Cvičení** Ukažte, že platí:

1. Operace $\| [R] \|_{\mathscr{M}}$ zachovává libovolné průniky, tj. platí

   je-li $\{M_i \mid i \in I\}$ libovolný systém podmnožin $V$, potom
$$\| [R] \|_{\mathscr{M}}(\bigcap_{i\in I} M_i) = \bigcap_{i\in I} \| [R] \|_{\mathscr{M}}(M_i).$$

2. Operace $\| [R] \|_{\mathscr{M}}$ obecně nezachovává konečná sjednocení, tj. nalezněte příklad modelu $\mathscr{M}$ a množin $M \subseteq V$, $N \subseteq V$ tak, že
$$\| [R] \|_{\mathscr{M}}(M \cup N) \neq \| [R] \|_{\mathscr{M}}(M) \cup \| [R] \|_{\mathscr{M}}(N).$$
   Platí alespoň jedna z inklusí?

3. Akce $R$ je *finitární* v $\mathscr{M}$, pokud je pro každý vrchol $v$ množina $\{v' \mid \text{existuje } v \xrightarrow{R} v'\}$ konečná.

   Systém $\{M_i \mid i \in I\}$ podmnožin $V$ je *filtrovaný*, pokud pro každých konečně mnoho množin $M_{i_1}, \ldots, M_{i_n}$ existuje $i \in I$ tak, že $M_{i_k} \subseteq M_i$ pro všechna $k \in \{1, \ldots, n\}$.

   Ukažte, že následující podmínky jsou pro model $\mathscr{M}$ ekvivalentní:

   (a) Akce $R$ je finitární v $\mathscr{M}$.

   (b) Operace $\| [R] \|_{\mathscr{M}}$ zachovává filtrovaná sjednocení, tj. platí
$$\| [R] \|_{\mathscr{M}}(\bigcup_{i\in I} M_i) = \bigcup_{i\in I} \| [R] \|_{\mathscr{M}}(M_i)$$
   pro libovolný filtrovaný systém $\{M_i \mid i \in I\}$ podmnožin $V$.

**7.7.9 Cvičení** Toto cvičení vyžaduje použití *Zornova lemmatu*. Zornovo lemma (nazývané také *princip maximality*) je následující tvrzení teorie množin:

> **(Zornovo lemma.)** *Ať $\langle X, \sqsubseteq\rangle$ je poset s následující vlastností:*
>
> > *Každý řetězec v $\langle X, \sqsubseteq\rangle$ má supremum.*
>
> *Potom v posetu $\langle X, \sqsubseteq\rangle$ pro libovolné $x \in X$ existuje maximální prvek $m$ takový, že $x \sqsubseteq m$.*

(*Maximální prvek v posetu* je takový prvek $m$, pro který neexistuje prvek $y$ takový, že $y \neq m$ a $m \sqsubseteq y$.)

Zornovo lemma je jedním z řady tvrzení, která nelze dokázat ze základní sady axiomů teorie množin. Tvrzení Zornova lemmatu lze tedy přijmout jako další axiom. (Připomeňme, že podobná situace je známa z axiomatické geometrie: jedná se o postulát o rovnoběžkách.) Podrobněji se o axiomatickém přístupu k teorii množin lze dočíst v knize [BaŠt86].

Pomocí Zornova lemmatu ukažte, že pro libovolný prvek $x$ Booleovy (modální) algebry $\mathbb{B}$ platí:

(*) jestliže $x \neq \bot$, potom existuje ultrafiltr $V$ na $\mathbb{B}$ takový, že $x \in V$.

K důkazu je dále zapotřebí následující pojem: řekneme, že množina $F$ prvků Booleovy algebry je *filtr*, pokud platí:

1. Jestliže $x \in F$ a $x \sqsubseteq y$, pak $y \in F$.
2. Jestliže $x \in F$ a $y \in F$, pak $(x \sqcap y) \in F$.

Pro důkaz tvrzení (*) postupujte podle těchto kroků:

1. Označte jako $\mathcal{F}$ množinu $\{F \mid F$ je filtr obsahující $x\}$.
2. Ukažte, že množina $\mathcal{F}$ je neprázdná.
   (Návod: jako $F_x$ označte množinu $\{y \mid x \sqsubseteq y\}$. Ukažte, že $F_x$ je filtr obsahující $x$.)
3. Uspořádejte množinu $\mathcal{F}$ následovně: $F_1 \leq F_2$ platí právě tehdy, když $F_1 \subseteq F_2$.
4. Ukažte, že v uspořádání $\leq$ má každý řetězec v $\mathcal{F}$ supremum.
5. Podle Zornova lemmatu vyvoďte, že v množině $\mathcal{F}$ je maximální prvek $V$ takový, že $F_x \leq V$.
6. Ukažte, že $V$ je ultrafiltr a že platí $x \in V$.

**7.7.10 Cvičení** Dokončete důkaz věty 7.6.2.

**7.7.11 Cvičení** Dokažte, že relace $\sim$ definovaná v konstrukci 7.6.3 je ekvivalence. Dále ukažte, že definice operací $^{-1}$, $\sqcap$, $\sqcup$, $[\![R]\!]$ jsou definovány korektně, tj. ukažte například, že platí:

> jestliže $\varphi \sim \psi$, pak $[\varphi]_\sim{}^{-1} = [\psi]_\sim{}^{-1}$.

Ukažte, že $\mathbb{K}$ je modální algebra typu $\Delta$.

**7.7.12 Cvičení** Promyslete, jak lze (v případě, že $At \neq \emptyset$) výsledky odstavce 7.6 modifikovat a dokázat tak větu 5.2.5.

Návod: postupujte podle následujících kroků.

1. Pracujte s modální logikou $Log_\Delta(At)$ a připomeňte rozšíření modalit z $\Delta$ na modality z $\Delta^*$ (viz cvičení 4.4.9).
2. Připomeňme, že deduktivní systém $\mathbf{KG}(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$ sestává ze stejných axiomů a odvozovacích pravidel jako systém $\mathbf{K}$ a jako jediný nový axiom je přidána formule $\langle\mathscr{R}\rangle[\mathscr{S}']\varphi \Rightarrow [\mathscr{S}]\langle\mathscr{R}'\rangle\varphi$, kde $\varphi$ je libovolná formule modální logiky $Log_\Delta(At)$.

3. Ukažte, že libovolný modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$ takový, že akční struktura $\mathbb{V}$ splňuje zobecněnou Churchovu-Rosserovu vlastnost $G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$ (viz definice 5.2.2), dává vzniknout modální algebře

$$\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}}, \| \neg \|_{\mathscr{M}}, \| \mathtt{tt} \|_{\mathscr{M}}, \| \mathtt{ff} \|_{\mathscr{M}} \| [\mathscr{R}] \| \rangle$$

typu $\Delta^*$, ve které platí rovnost

$$\| \langle \mathscr{R} \rangle \|_{\mathscr{M}} \| [\mathscr{S}'] \|_{\mathscr{M}}(M) \cap \| [\mathscr{S}] \|_{\mathscr{M}} \| \langle \mathscr{R}' \rangle \|_{\mathscr{M}}(M) = \| \langle \mathscr{R} \rangle \|_{\mathscr{M}} \| [\mathscr{S}'] \|_{\mathscr{M}}(M)$$

pro libovolnou množinu $M \subseteq V$.

4. Připomeňte, že $\Lambda G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$ značí třídu všech akčních struktur, ve kterých je splněna zobecněná Churchova-Rosserova vlastnost $G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$.

5. Definujte: modální algebra $\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![\mathscr{R}]\!] \rangle$ typu $\Delta^*$ *splňuje zobecněnou Churchovu-Rosserovu vlastnost* $G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$, pokud pro každé $x \in B$ platí rovnost

$$\langle\!\langle \mathscr{R} \rangle\!\rangle [\![\mathscr{S}']\!](x) \sqcap [\![\mathscr{S}]\!] \langle\!\langle \mathscr{R}' \rangle\!\rangle(x) = \langle\!\langle \mathscr{R} \rangle\!\rangle [\![\mathscr{S}']\!](x).$$

6. Ukažte analogii věty 7.6.1:

   *Pro libovolnou formuli $\varphi$ jsou následující podmínky ekvivalentní:*

   1. $\models_{\Lambda G(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')} \varphi$
   2. *Pro libovolnou modální algebru $\mathbb{B}$ splňující zobecněnou Churchovu-Rosserovu vlastnost a libovolné $\mathbb{B}$-ohodnocení $u$ platí $u(\varphi) = \top$.*

7. Ukažte analogii věty 7.6.2:

   *Jestliže $\mathbf{KG}(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}') \vdash \varphi$, potom $u(\varphi) = \top$ pro libovolnou modální algebru*

   $$\mathbb{B} = \langle B, \sqcap, \sqcup, ^{-1}, \top, \bot, [\![\mathscr{R}]\!] \rangle$$

   *splňující zobecněnou Churchovu-Rosserovu vlastnost a libovolné $\mathbb{B}$-ohodnocení $u$.*

8. Pro deduktivní systém $\mathbf{KG}(\mathscr{R}, \mathscr{S}', \mathscr{S}, \mathscr{R}')$ sestrojte Lindenbaumovu-Tarského algebru podobně jako v konstrukci 7.6.3. Ukažte, že tato modální algebra splňuje zobecněnou Churchovu-Rosserovu vlastnost.

9. Dokažte větu o úplnosti 5.2.5.

# Kapitola 8

# Modální $\mu$-kalkulus

Zavedení modálního $\mu$-kalkulu je vynuceno malou vyjdřovací schopností modální výrokové logiky.

Připomeňme některé nevýhody jazyka modální logiky:

1. Existují přirozeně vyjádřené vlastnosti akčních struktur, které nejsou modálně definovatelné (viz odstavec 5.3).
2. Pro obecný program $p$ jazyka w*hile* programů neumíme najít nejslabší liberální prepodmínku — viz odstavec 6.4.

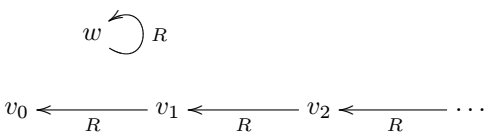
Syntaxi jazyka modální výrokové logiky v této kapitole obohatíme. Budeme tak mít k dispozici více formulí a naše vyjadřovací schopnost se tak pronikavě zvýší.

Zavedení a funkci modálního $\mu$-kalkulu budeme v této kapitole ilustrovat právě nejslabší liberální prepodmínkou pro obecný w*hile* cyklus. Nejprve ukážeme, že modální logika není schopna vystihnout schopnost nekonečného chování v akční struktuře.

## 8.1 Neadekvátnost modální logiky

Pro zavedení modálního $\mu$-kalkulu bude výhodné nahlížet na formule jako na extense: přesněji v modelu $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$ budeme danou formuli $\varphi$ ztotožňovat s množinou vrcholů $\| \varphi \|_{\mathscr{M}}$, ve kterých je $\varphi$ $\mathscr{M}$-vynucena. Formule $\varphi$ se tak stane predikátem na množině vrcholů akční struktury $\mathbb{V}$. Tohoto pohledu na modální formule jsme využili v kapitole 7. Připomeňme, že ne každá množina vrcholů je extensí nějaké modální formule — viz příklad 7.2.4. Uvedeme ještě jeden, zajímavější příklad.

**8.1.1 Příklad** Uvažujme o akční struktuře $\mathbb{V}$ z následujícího obrázku:




Ve struktuře $\mathbb{V}$ jsou vrcholy dvou typů: ty, které mohou donekonečna vykonávat akci $R$ (takový vrchol je jediný, a to $w$) a vrcholy, které tuto vlastnost nemají (jsou to všechny vrcholy $v_n$). Ukážeme, že tuto vlastnost nelze vystihnout modální formulí, která neobsahuje atomickou formuli. Přesněji: ukážeme, že pro libovolný modální model $\mathscr{M}$ a pro libovolnou modální formuli $\varphi$ neobsahující atomické formule, platí:

$$\text{Jestliže } w \Vdash_{\mathscr{M}} \varphi, \text{ potom existuje } n_0 \text{ tak, že platí } v_n \Vdash_{\mathscr{M}} \varphi \text{ pro všechna } n \geq n_0.$$

Protože předpokládáme, že formule $\varphi$ neobsahuje atomické formule, můžeme předpokládat, že $\varphi$ je v jednom z následujících tvarů (viz cvičení 4.4.3):

- $\varphi$ je `tt`. Tvrzení je triviální, zvolte $n_0 = 0$.
- $\varphi$ je `ff`. I v tomto případě je tvrzení je triviální, zvolte $n_0 = 0$.

- $\varphi$ je $\varphi_1 \wedge \varphi_2$. Jako indukční předpoklad předpokládejme platnost tvrzení pro $\varphi_1$ a $\varphi_2$:

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1$, potom existuje $n_1$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1$ pro všechna $n \geq n_1$.
  Jestliže $w \Vdash_{\mathscr{M}} \varphi_2$, potom existuje $n_2$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_2$ pro všechna $n \geq n_2$.

  Potom samozřejmě platí

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1 \wedge \varphi_2$, potom existuje $n_0$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1 \wedge \varphi_2$ pro všechna $n \geq n_0$.

  kde jako číslo $n_0$ zvolíme maximum z čísel $n_1$ a $n_2$.

- $\varphi$ je $\varphi_1 \vee \varphi_2$. Jako indukční předpoklad předpokládejme platnost tvrzení pro $\varphi_1$ a $\varphi_2$:

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1$, potom existuje $n_1$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1$ pro všechna $n \geq n_1$.
  Jestliže $w \Vdash_{\mathscr{M}} \varphi_2$, potom existuje $n_2$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_2$ pro všechna $n \geq n_2$.

  Potom platí

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1 \vee \varphi_2$, potom existuje $n_0$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1 \vee \varphi_2$ pro všechna $n \geq n_0$.

  kde jako číslo $n_0$ zvolíme minimum z čísel $n_1$ a $n_2$.

- $\varphi$ je $[R]\varphi_1$. Jako indukční předpoklad předpokládejme platnost tvrzení pro $\varphi_1$:

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1$, potom existuje $n_1$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1$ pro všechna $n \geq n_1$.

  Předpokládáme-li, že $w \Vdash_{\mathscr{M}} [R]\varphi_1$, znamená to, že $w \Vdash_{\mathscr{M}} \varphi_1$. Podle indukčního předpokladu je nyní možné zvolit $n_0 = n_1$.

- $\varphi$ je $\langle R\rangle\varphi_1$. Jako indukční předpoklad předpokládejme platnost tvrzení pro $\varphi_1$:

  Jestliže $w \Vdash_{\mathscr{M}} \varphi_1$, potom existuje $n_1$ tak, že platí $v_n \Vdash_{\mathscr{M}} \varphi_1$ pro všechna $n \geq n_1$.

  Předpokládáme-li, že $w \Vdash_{\mathscr{M}} \langle R\rangle\varphi_1$, znamená to, že $w \Vdash_{\mathscr{M}} \varphi_1$. Podle indukčního předpokladu je nyní možné zvolit $n_0 = n_1$.

Právě dokázané tvrzení má tento důsledek: jestliže v nějakém modálním modelu $\mathscr{M}$ na struktuře $\mathbb{V}$ je modální formule $\varphi$ vynucena pouze ve vrcholu $w$, musí $\varphi$ obsahovat alespoň jednu atomickou formuli.

Bez použití atomických formulí tedy nelze v modálních modelech na struktuře rozlišit možnost nekonečného chování od nemožnosti nekonečného chování.

Atomickou formulí samozřejmě vrchol $w$ odlišit lze, uvažujte o valuaci $\mathscr{V}(a) = \{w\}$. Potom $w$ je jediný vrchol, ve kterém je vynucena formule $a$.

Předchozí příklad ukazuje, že Hennessyho-Milnerova modální logika není schopna postihnout možnost nekonečného chování v akční struktuře. Zavedením modálního $\mu$-kalkulu se této nepříjemnosti zbavíme — viz cvičení 8.8.12.

## 8.2 Ještě jednou nejslabší liberální prepodmínka

Příkladem náhledu na formuli jako na extensi byla definice nejslabší liberální prepodmínky `while` programu $p$ vzhledem k formuli $\varphi$ — viz definice 6.4.2. Připomeňme, že pro formuli bez modalit $\varphi$ a program jazyka `while` programů $p$ je formule bez modalit $\mathsf{wlp}(\varphi \mid p)$ *nejslabší liberální prepodmínkou programu* $p$ *vzhledem k* $\varphi$, pokud pro každou paměť $M$ platí

$M \Vdash_{\mathscr{M}} \mathsf{wlp}(\varphi \mid p)$ právě tehdy, když $M \Vdash_{\mathscr{M}} [p]\varphi$.

Označme jako $H$ množinu $\{M \mid M \Vdash_{\mathscr{M}} [p]\varphi\}$. Podle definice je množina $H$ přesně extensí formule $[p]\varphi$ v zamýšleném modálním modelu $\mathscr{M}$ jazyka `while` programů.

Ukázat existenci nejslabší liberální prepodmínky $p$ vzhledem k $\varphi$ znamenalo tedy nalézt formuli bez modalit, jejíž extense je také množina $H$. V tvrzení 6.4.5 jsme ukázali, že nejslabší liberální prepodmínku lze nalézt pro každý program, neobsahující cyklus `while`.

Pro programy používající cyklus `while` je hledání nejslabší liberální prepodmínky obtížný problém. V následujícím tvrzení ukážeme, jaké vlastnosti musí množina

$$H = \{M \mid M \Vdash_{\mathscr{M}} [p]\varphi\}$$

pro program $p$ tvaru `while` $b$ `do` $q$ `od` splňovat.

**8.2.1 Lemma** *Označme $H = \{M \mid M \Vdash_{\mathscr{M}} [p]\varphi\}$. Potom platí následující dvě tvrzení:*

1. *Platí rovnost*
$$H = \Big(\| [q] \|_{\mathscr{M}}(H) \cap \| b \|_{\mathscr{M}}\Big) \cup \Big(\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}\Big).$$
2. *Jestliže $X$ je množina pamětí taková, že platí rovnost*
$$X = \Big(\| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}}\Big) \cup \Big(\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}\Big)$$
*potom platí $X \subseteq H$.*

Důkaz. Důkaz prvního tvrzení: dokážeme nejprve inklusi

$$H \subseteq \Big(\| [q] \|_{\mathscr{M}}(H) \cap \| b \|_{\mathscr{M}}\Big) \cup \Big(\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}\Big).$$

Předpokládejme, že máme paměť $M \in H$. Potom víme, že platí $M \Vdash_{\mathscr{M}} [p]\varphi$. Může nastat právě jeden ze dvou případů:

1. $M \in \| b \|_{\mathscr{M}}$. V tomto případě by stačilo ukázat, že $M \in \| [q] \|_{\mathscr{M}}(H)$.

   Zvolme libovolnou akci $M \xrightarrow{q} M'$. Podle definice extense boxu stačí ukázat, že platí $M' \in H$, neboli, že platí $M' \Vdash_{\mathscr{M}} [p]\varphi$.

   Zvolme libovolnou akci $M' \xrightarrow{p} M''$. Podle definice tvorby akcí v modálním modelu $\mathscr{M}$ potom máme akci $M \xrightarrow{p} M''$. Protože platí $M \Vdash_{\mathscr{M}} [p]\varphi$, dostáváme platnost vztahu $M'' \Vdash_{\mathscr{M}} \varphi$. Ukázali jsme, že platí $M' \Vdash_{\mathscr{M}} [p]\varphi$.

2. $M \notin \| b \|_{\mathscr{M}}$. Potom podle definice extense spojky negace platí $M \in \| \neg b \|_{\mathscr{M}}$. Stačí tedy ukázat, že $M \in \| \varphi \|_{\mathscr{M}}$. Protože ale cyklus $p$ z paměti $M$ vůbec neproběhne a protože platí $M \Vdash_{\mathscr{M}} [p]\varphi$, dostáváme, že platí $M \Vdash_{\mathscr{M}} \varphi$.

Důkaz inkluse $\Big(\| [q] \|_{\mathscr{M}}(H) \cap \| b \|_{\mathscr{M}}\Big) \cup \Big(\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}\Big) \subseteq H$ opět rozdělíme na dvě části.

1. Ať $M \in \| [q] \|_{\mathscr{M}}(H) \cap \| b \|_{\mathscr{M}}$. Abychom ukázali, že $M \in H$, stačí ukázat, že $M \Vdash_{\mathscr{M}} [p]\varphi$.

   Zvolme libovolnou akci $M \xrightarrow{p} M'$. Díky tomu, že platí $M \Vdash_{\mathscr{M}} b$, existuje paměť $M_1$ a akce $M \xrightarrow{q} M_1$ a $M_1 \xrightarrow{p} M'$.

   Protože $M \in H$, je podle definice extense boxu $M_1 \in H$, neboli platí $M_1 \Vdash_{\mathscr{M}} [p]\varphi$. Ukázali jsme, že platí $M \Vdash_{\mathscr{M}} [p]\varphi$.

2. Ať $M \in \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$. Abychom ukázali, že $M \in H$, stačí ukázat, že $M \Vdash_{\mathscr{M}} [p]\varphi$.

   Zvolme libovolnou akci $M \xrightarrow{p} M'$. Protože $M \Vdash_{\mathscr{M}} \neg b$, platí $M = M'$ a tudíž $M \Vdash_{\mathscr{M}} [p]\varphi$.

Důkaz druhého tvrzení: zvolme libovolnou paměť $M \in X$. Chceme ukázat, že platí $M \Vdash_{\mathscr{M}} [p]\varphi$. Zvolme libovolnou akci $M \xrightarrow{p} M'$. Máme ukázat platnost vztahu $M' \Vdash_{\mathscr{M}} \varphi$. Budeme postupovat indukcí podle výšky stromu, který akci $M \xrightarrow{p} M'$ odvozuje.

1. Výška stromu, odvozujícího akci $M \xrightarrow{p} M'$ je 1. To znamená, že cyklus **while** vůbec neproběhne a že platí $M \Vdash_{\mathscr{M}} \neg b$ a $M = M'$.

   V tomto případě je $M \in \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$, protože množiny $\| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}}$ a $\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$ jsou disjunktní.

   Ukázali jsme, že $M' \Vdash_{\mathscr{M}} \varphi$, neboli platí $M \in H$.

2. V případě, že výška stromu, odvozujícího akci $M \xrightarrow{p} M'$, je $n+1$, existují akce $M \xrightarrow{q} M_1$ a $M_1 \xrightarrow{p} M'$. Navíc platí $M \Vdash_{\mathscr{M}} b$.

   Víme tedy, že platí $M \in \| [q] \|_{\mathscr{M}}(H) \cap \| b \|_{\mathscr{M}}$, protože množina $\| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}}$ je disjunktní s množinou $\| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$.

   Pak ale z definice extense boxu je $M_1 \in X$, protože $M \in \| [q] \|_{\mathscr{M}}(X)$.

   Nyní můžeme použít indukční předpoklad: výška stromu odvozujícího akci $M_1 \xrightarrow{p} M'$ je $n$. Podle indukčního přepokladu je tedy $M_1 \in H$, tudíž $M' \Vdash_{\mathscr{M}} \varphi$.

■

Rovnost pro množinu $H$ z lemmatu 8.2.1 lze interpretovat také tak, že množina $H$ je *řešením jisté rovnice*. Definujme funkci $f$ následujícím způsobem:

Funkce $f$ přiřadí množině pamětí $X$ množinu pamětí $\Big( \| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}} \Big) \cup \Big( \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}} \Big)$.

První tvrzení lemmatu 8.2.1 říká, že platí rovnost $H = f(H)$, neboli, že *množina $H$ je řešením rekursivní rovnice $X = f(X)$*. Druhé tvrzení lemmatu 8.2.1 říká, že $H$ je *největším řešením rovnice $X = f(X)$ vzhledem k inklusi*, protože jsme ukázali, že když platí $X = f(X)$, potom je $X$ nutně podmnožinou $H$.

Postup při hledání extense nejslabší liberální prepodmínky, tj. výše popsané množiny $H$, pro **while** cyklus lze tedy shrnout v následujících bodech:

1. Sestavíme funkci $f$, která přiřadí množině pamětí $X$ množinu pamětí

$$\Big( \| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}} \Big) \cup \Big( \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}} \Big).$$

2. Budeme muset umět dokázat, že existuje řešení rovnice $X = f(X)$, tj. že funkce $f$ má alespoň jeden *pevný bod*. Dále budeme muset ukázat, že funkce $f$ má *největší pevný bod vzhledem k inklusi*.

   Množina $H$ je pak *největším pevným bodem funkce $f$ vzhledem k inklusi*.

3. Pro množinu $H$ budeme muset nalézt formuli bez modalit $\alpha$ takovou, že množina $H$ je extense formule $\alpha$.

Tyto tři body velmi dobře ilustrují, o co v $\mu$-kalkulu jde. Probereme je tedy znova a obecněji.

1. Namísto speciální funkce $f$ bychom měli studovat obecné funkce, které množině vrcholů nějaké akční struktury přiřadí opět množinu vrcholů této akční struktury. Řečeno jinak:

   *Pro obecnou akční strukturu $\mathbb{V}$ budeme studovat funkce $f : exp(V) \to exp(V)$.*

2. U funkce $f : exp(V) \to exp(V)$ budeme hledat pevné body, tj. řešení rovnice $X = f(X)$. Pro řadu funkcí bude alespoň jedno řešení existovat: množina $exp(V)$ je částečně uspořádaná inklusí, jde dokonce o úplný svaz. Z teorie pevných bodů *monotónních* funkcí na úplném svazu je dobře známo, že řešení rovnice $X = f(X)$ vždy existuje — využijeme toho. Dokonce lze ukázat, že mezi řešeními rovnice $X = f(X)$ vždy existují dvě prominentní řešení: největší a nejmenší. Jinými slovy:

   *Řešení rovnice $X = f(X)$ existuje pro monotónní $f$. Těchto řešení může být více. Vždy však lze nalézt nejmenší a největší z nich.*

3. Budeme potřebovat, aby největší a nejmenší řešení rovnice $X = f(X)$ byla vždy extensemi nějakých formulí. V příkladu 7.2.4 jsme však viděli, že ne každá množina formulí musí být (v pevném modelu) extensí nějaké formule. Tuto obtíž překleneme elegantním způsobem:

*Do syntaxe formulí Hennessyho-Milnerovy modální logiky přidáme nové formule tak, aby bylo zaručeno, že největší a nejmenší řešení rovnice $X = f(X)$ bylo vždy extensí nějaké formule.*

Právě jsme nastínili důvody zavedení modálního $\mu$-kalkulu. V dalším textu této kapitoly již budeme pouze zpřesňovat tři výše uvedené myšlenky.

Začneme vyšetřováním pevných bodů monotónních zobrazení.

## 8.3 Pevné body v úplných svazech

Připomeňme, že úplný svaz je takový poset $\langle X, \sqsubseteq\rangle$, kde *každá* podmnožina $X$ má supremum a infimum. Následující věta umožňuje najít řešení rovnice $f(x) = x$ v případě, že $\langle X, \sqsubseteq\rangle$ je úplný svaz a $f$ je monotónní zobrazení z $\langle X, \sqsubseteq\rangle$ do $\langle X, \sqsubseteq\rangle$.

**8.3.1 Věta (Tarski)** *Ať $\mathbb{X} = \langle X, \sqsubseteq\rangle$ je úplný svaz, ať $f : \mathbb{X} \to \mathbb{X}$ je monotónní zobrazení. Pak platí:*

1. *Zobrazení $f$ má nejmenší pevný bod, tj. existuje prvek $\mu f \in X$ takový, že*
   (a) $f(\mu f) = \mu f$
   (b) *Pro libovolné $x \in X$ takové, že $f(x) = x$, platí $\mu f \sqsubseteq x$.*

   *Pro toto $\mu f$ navíc platí: $\mu f = \inf\{x \in X \mid f(x) \sqsubseteq x\}$.*
2. *Zobrazení $f$ má největší pevný bod, tj. existuje prvek $\nu f \in X$ takový, že*
   (a) $f(\nu f) = \nu f$
   (b) *Pro libovolné $x \in X$ takové, že $f(x) = x$, platí $x \sqsubseteq \nu f$.*

   *Pro toto $\nu f$ navíc platí: $\nu f = \sup\{x \in X \mid x \sqsubseteq f(x)\}$.*

Důkaz. Dokážeme pouze tvrzení o nejmenším pevném bodu, pro důkaz druhého tvrzení odkazujeme na cvičení 8.8.1.

Označme množinu $\{x \in X \mid f(x) \sqsubseteq x\}$ jako $M$. Protože předpokládáme, že $\langle X, \sqsubseteq\rangle$ je úplný svaz, infimum $\inf M$ jistě existuje. Označme jej $\mu f$.

Nejprve ukážeme platnost rovnosti $f(\mu f) = \mu f$.

- Platí $f(\mu f) \sqsubseteq \mu f$. Z definice infima totiž dostáváme, že $\mu f \sqsubseteq x$ pro všechna $x \in M$. Protože $f$ je monotónní, platí $f(\mu f) \sqsubseteq f(x)$ pro všechna $x \in M$. Pro každý prvek $x$ množiny $M$ ale platí $f(x) \sqsubseteq x$. Tudíž platí $f(\mu f) \sqsubseteq x$ pro všechna $x \in M$. Prvek $f(\mu f)$ je proto dolní odhad množiny $M$. Infimum je ale největší dolní odhad a proto platí nerovnost $f(\mu f) \sqsubseteq \mu f$.
- Platí $\mu f \sqsubseteq f(\mu f)$. V předchozím bodu jsme vlastně ukázali, že $\mu f \in M$. Protože $f$ je monotónní, je i $f(\mu f) \in M$. Nerovnost $\mu f \sqsubseteq f(\mu f)$ tedy plyne z toho, že $\mu f$ je dolní odhad množiny $M$.

Předpokládejme, že $x$ je takové, že $f(x) = x$. Pak $x \in M$, a nerovnost $\mu f \sqsubseteq x$ plyne z toho, že $\mu f$ je dolní odhad množiny $M$. ■

Předchozí věta ukazuje, že pro dané monotónní zobrazení $f : \mathbb{X} \to \mathbb{X}$ budou mít množiny

$$\{x \in X \mid f(x) \sqsubseteq x\} \quad \text{a} \quad \{x \in X \mid x \sqsubseteq f(x)\}$$

zásadní význam. Zavedeme proto následující terminologii.

**8.3.2 Definice** Ať $f : \mathbb{X} \to \mathbb{X}$ je monotónní zobrazení. Řekneme, že $x \in X$ je:

1. *Prefixní bod zobrazení $f$*, pokud platí $f(x) \sqsubseteq x$.
2. *Postfixní bod zobrazení $f$*, pokud platí $x \sqsubseteq f(x)$.
3. *Pevný bod zobrazení $f$*, pokud platí $x = f(x)$.

Množinu všech prefixních bodů zobrazení $f$ označíme $\mathsf{Pre}(f)$, množinu všech postfixních bodů zobrazení $f$ označíme $\mathsf{Post}(f)$, a množinu všech pevných bodů zobrazení $f$ označíme $\mathsf{Fix}(f)$.

Tarského větu lze nyní přeformulovat následovně: Pro monotónní zobrazení $f : \mathbb{X} \to \mathbb{X}$ na úplném svazu platí:

1. Zobrazení $f$ má nejmenší pevný bod $\mu f = \inf \mathsf{Pre}(f)$.
2. Zobrazení $f$ má největší pevný bod $\nu f = \sup \mathsf{Post}(f)$.

Při motivačních úvahách v odstavci 8.2 jsme uvažovali o zobrazeních $f : exp(V) \to exp(V)$, přiřazujících množině vrcholů akční struktury opět množinu vrcholů akční struktury. Množina $exp(V)$ je poset: částečné uspořádání je zde dáno inklusí. Tento poset je úplný svaz: vezměme libovolnou množinu $M \subseteq exp(V)$. Připomeneme, jak najít supremum a infimum množiny $M$.

Jestliže $M \subseteq exp(V)$, znamená to, že každý prvek množiny $M$ je podmnožinou množiny $V$. Množinu $M$ si tedy můžeme představit jako systém $\{M_i \mid i \in I\}$, kde každé $M_i$ je podmnožina množiny $V$.

- Supremum $\{M_i \mid i \in I\}$ v $\langle exp(V), \subseteq\rangle$ je množinové sjednocení $\bigcup_{i\in I} M_i$.
- Infimum $\{M_i \mid i \in I\}$ v $\langle exp(V), \subseteq\rangle$ je množinový průnik $\bigcap_{i\in I} M_i$.

Uvedeme příklady některých monotónních zobrazení z $\langle exp(V), \subseteq\rangle$ do $\langle exp(V), \subseteq\rangle$.

**8.3.3 Příklad** *Konstantní zobrazení je monotónní.* Zvolme pevně množinu $P \subseteq V$. Zobrazení $f_P : exp(V) \to exp(V)$ definujme předpisem $f_P(M) = P$ pro libovolné $M \subseteq V$. Jestliže platí $M_1 \subseteq M_2$, potom zřejmě platí $f_P(M_1) \subseteq f_P(M_2)$.

*Průnik s pevnou množinou je monotónní.* Zvolme pevně množinu $P \subseteq V$. Zobrazení $g_P : exp(V) \to exp(V)$ definujme předpisem $g_P(M) = P \cap M$ pro libovolné $M \subseteq V$. Jestliže platí $M_1 \subseteq M_2$, potom zřejmě platí $g_P(M_1) \subseteq g_P(M_2)$, protože $P \cap M_1 \subseteq P \cap M_2$.

*Sjednocení s pevnou množinou je monotónní.* Zvolme pevně množinu $P \subseteq V$. Zobrazení $h_P : exp(V) \to exp(V)$ definujme předpisem $h_P(M) = P \cup M$ pro libovolné $M \subseteq V$. Jestliže platí $M_1 \subseteq M_2$, potom zřejmě platí $h_P(M_1) \subseteq h_P(M_2)$, protože $P \cup M_1 \subseteq P \cup M_2$.

*Operace množinového doplňku není monotónní.* Zobrazení $h : exp(V) \to exp(V)$ definujme předpisem $h(M) = V \setminus M$ pro libovolné $M \subseteq V$. Z toho, že platí $M_1 \subseteq M_2$, však nelze usoudit na platnost vztahu $V \setminus M_1 \subseteq V \setminus M_2$.

V odstavci 7.3 jsme poukázali na to, že na modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ Hennessyho-Milnerovy modální logiky lze pohlížet jako na modální algebru

$$\langle exp(V), \| \wedge \|_{\mathscr{M}}, \| \vee \|_{\mathscr{M}}, \| \neg \|_{\mathscr{M}}, \| \mathtt{tt} \|_{\mathscr{M}}, \| \mathtt{ff} \|_{\mathscr{M}}, \| [R] \|_{\mathscr{M}}\rangle$$

Modální algebry byly zavedeny v kapitole 7 kvůli důkazu věty o úplnosti systému **K** minimální modální logiky. Definice modální algebry vznikla přirozeným obohacením struktury Booleovy algebry podmnožin vrcholů akční struktury o extense modality box.

Dalším příkladem zobrazení z $\langle exp(V), \subseteq\rangle$ do $\langle exp(V), \subseteq\rangle$ pro modální model $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$ je tedy extense boxu: zobrazení $\| [R] \|_{\mathscr{M}}$ přiřazuje množině $M \subseteq V$ množinu

$$\| [R] \|_{\mathscr{M}}(M) = \{v \in V \mid \text{je-li } v \xrightarrow{R} v',\ \text{pak } v' \in M\}$$

Povšimněme si, že v definici extense boxu jsme valuaci $\mathscr{V}$ vůbec nevyužili. Můžeme tedy o extensi boxu akce $R$ mluvit pouze ve smyslu akční struktury $\mathbb{V}$:

$$\| [R] \|_{\mathbb{V}}(M) = \{v \in V \mid \text{je-li } v \xrightarrow{R} v',\ \text{pak } v' \in M\}$$

Podobně na volbě modelu $\langle \mathbb{V}, \mathscr{V}\rangle$, tj. na konkrétní volbě valuace $\mathscr{V}$, nezávisely definice extense logických spojek z výrokové logiky $\wedge$, $\vee$, $\neg$ a formulí `tt` a `ff`. Pro libovolný model na $\mathbb{V}$ jsou vždy definovány jako množinové operace $\cap$, $\cup$, $V \setminus$ a jako množiny $V$ a $\emptyset$.

Abstrahujeme-li tedy od jednotlivých modálních modelů na konkrétní akční struktuře $\mathbb{V}$, dostaneme vždy modální algebru, pokud definujeme extense logických spojek následovně:

**8.3.4 Definice** Pro akční strukturu $\mathbb{V}$ typu $\Delta$ definujeme:

1. *Extense spojky* $\wedge$: jde o zobrazení $\| \wedge \|_{\mathbb{V}}$, které dvojici množin $(M, N)$ přiřadí jejich průnik $M \cap N$.
2. *Extense spojky* $\vee$: jde o zobrazení $\| \vee \|_{\mathbb{V}}$, které dvojici množin $(M, N)$ přiřadí jejich sjednocení $M \cup N$.
3. *Extense spojky* $\neg$: jde o zobrazení $\| \neg \|_{\mathbb{V}}$, které množině množin $M$ přiřadí její doplněk $V \setminus M$.
4. *Extense formule* `tt`: množina $\| \mathtt{tt} \|_{\mathbb{V}}$ je definována jako množina $V$.
5. *Extense formule* `ff`: množina $\| \mathtt{ff} \|_{\mathbb{V}}$ je definována jako prázdná množina $\emptyset$.
6. *Extense boxu* $R$: zobrazení $\| [R] \|_{\mathbb{V}}$ přiřazuje množině vrcholů $M$ množinu

$$\| [R] \|_{\mathbb{V}}(M) = \{v \in V \mid \text{je-li } v \xrightarrow{R} v', \text{ pak } v' \in M\}.$$

Ukážeme, že získaná struktura je modální algebra.

**8.3.5 Lemma** *Ať $\mathbb{V}$ je akční struktura typu $\Delta$. Potom*

$$\langle exp(V), \| \wedge \|_{\mathbb{V}}, \| \vee \|_{\mathbb{V}}, \| \neg \|_{\mathbb{V}}, \| \mathtt{tt} \|_{\mathbb{V}}, \| \mathtt{ff} \|_{\mathbb{V}}, \| [R] \|_{\mathbb{V}} \rangle$$

*je modální algebra typu $\Delta$.*

Důkaz. Je triviální ukázat, že $\langle exp(V), \| \wedge \|_{\mathbb{V}}, \| \vee \|_{\mathbb{V}}, \| \neg \|_{\mathbb{V}}, \| \mathtt{tt} \|_{\mathbb{V}}, \| \mathtt{ff} \|_{\mathbb{V}} \rangle$ je Booleova algebra. Abychom ukázali, že přidáním operací $\| [R] \|_{\mathbb{V}}$ vznikne modální algebra, stačí ukázat, že platí rovnost $\| [R] \|_{\mathbb{V}}(V) = V$ a že pro libovolné $M \subseteq V$, $N \subseteq V$ platí $\| [R] \|_{\mathbb{V}}(M \cap N) = \| [R] \|_{\mathbb{V}}(M) \cap \| [R] \|_{\mathbb{V}}(N)$. Tyto rovnosti plynou bezprostředně z důkazu tvrzení 7.3.1 — ani tam jsme totiž konkrétní valuace nevyužili. ■

**8.3.6 Lemma** $\| [R] \|_{\mathbb{V}}$ *je monotónní zobrazení.*

Důkaz. Vezměme množiny $M_1$ a $M_2$ tak, že $M_1 \subseteq M_2$. Abychom ukázali, že platí $\| [R] \|_{\mathbb{V}}(M_1) \subseteq \| [R] \|_{\mathbb{V}}(M_2)$, stačí ukázat, že platí rovnost

$$\| [R] \|_{\mathbb{V}}(M_1) \cap \| [R] \|_{\mathbb{V}}(M_2) = \| [R] \|_{\mathbb{V}}(M_1).$$

Tato rovnost ale plyne z toho, že $M_1 \cap M_2 = M_1$ a z toho, že $\| [R] \|_{\mathbb{V}}$ zachovává konečné průniky. ■

**8.3.7 Příklad** Podle Tarského věty 8.3.1 má každé monotónní zobrazení z $\langle exp(V), \subseteq \rangle$ do $\langle exp(V), \subseteq \rangle$ největší a nejmenší pevný bod. Podívejme se, jak tyto největší a nejmenší body vypadají pro monotónní zobrazení z příkladu 8.3.3 a lemmatu 8.3.6.

1. Konstantní zobrazení $f_P$. Toto zobrazení má samozřejmě jediný pevný bod, a sice množinu $P$. Tento pevný bod je samozřejmě nejmenším pevným bodem a největším pevným bodem současně. Ukažme, jak to lze nahlédnout i z tvrzení Tarského věty.
   (a) Nejmenší pevný bod. Množina $M \subseteq V$ je prefixním bodem $f_P$ právě tehdy, když platí $f_P(M) \subseteq M$. To nastává právě tehdy, když $P \subseteq M$. Je tedy $\mathsf{Pre}(f_P) = \{M \mid P \subseteq M\}$. Nejmenší pevný bod je infimum množiny všech prefixních bodů, tj. $\mu f_P = \bigcap \{M \mid P \subseteq M\}$, protože infima jsou množinové průniky. Protože $\bigcap \{M \mid P \subseteq M\} = P$, je množina $P$ nejmenším pevným bodem zobrazení $f_P$.
   (b) Největší pevný bod. Množina $M \subseteq V$ je postfixním bodem $f_P$ právě tehdy, když platí $M \subseteq f_P(M)$. To nastává právě tehdy, když $M \subseteq P$. Je tedy $\mathsf{Post}(f_P) = \{M \mid M \subseteq P\}$. Největší pevný bod je supremum množiny všech postfixních bodů, tj. $\nu f_P = \bigcup \{M \mid M \subseteq P\}$, protože suprema jsou množinová sjednocení. Protože $\bigcup \{M \mid M \subseteq P\} = P$, je množina $P$ největším pevným bodem zobrazení $f_P$.
2. Průnik s pevnou množinou — zobrazení $g_P$.

(a) Nejmenší pevný bod. Množina $M \subseteq V$ je prefixním bodem $g_P$ právě tehdy, když platí $g_P(M) \subseteq M$. To nastává právě tehdy, když $P \cap M \subseteq M$. Inkluse $P \cap M \subseteq M$ je však splněna pro libovolnou množinu $M$. Je tedy $\mathsf{Pre}(g_P) = \{M \mid M \subseteq V\}$. Nejmenší pevný bod je infimum množiny všech prefixních bodů, tj. $\mu g_P = \bigcap \{M \mid M \subseteq V\}$, protože infima jsou množinové průniky. Protože $\bigcap \{M \mid M \subseteq V\} = \emptyset$, je prázdná množina nejmenším pevným bodem zobrazení $g_P$.

(b) Největší pevný bod. Množina $M \subseteq V$ je postfixním bodem $g_P$ právě tehdy, když platí $M \subseteq g_P(M)$. To nastává právě tehdy, když $M \subseteq P \cap M$. Inkluse $M \subseteq P \cup M$ je však splněna přesně tehdy, když platí $M \subseteq P$.

Je tedy $\mathsf{Post}(g_P) = \{M \mid M \subseteq P\}$. Největší pevný bod je supremum množiny všech postfixních bodů, tj. $\nu g_P = \bigcup \{M \mid M \subseteq P\}$, protože suprema jsou množinová sjednocení. Protože $\bigcup \{M \mid M \subseteq P\} = P$, je množina $P$ největším pevným bodem zobrazení $g_P$.

3. Sjednocení s pevnou množinou — zobrazení $h_P$.

(a) Nejmenší pevný bod. Množina $M \subseteq V$ je prefixním bodem $h_P$ právě tehdy, když platí $h_P(M) \subseteq M$. To nastává právě tehdy, když $P \cup M \subseteq M$. Inkluse $P \cup M \subseteq M$ je však splněna práě tehdy, když $P \subseteq M$. Je tedy $\mathsf{Pre}(h_P) = \{M \mid P \subseteq M\}$. Nejmenší pevný bod je infimum množiny všech prefixních bodů, tj. $\mu h_P = \bigcap \{M \mid P \subseteq M\}$, protože infima jsou množinové průniky. Protože $\bigcap \{M \mid P \subseteq M\} = P$, je množina $P$ nejmenším pevným bodem zobrazení $h_P$.

(b) Největší pevný bod. Množina $M \subseteq V$ je postfixním bodem $h_P$ právě tehdy, když platí $M \subseteq h_P(M)$. To nastává právě tehdy, když $M \subseteq P \cup M$. Inkluse $M \subseteq P \cup M$ je však splněna vždy. Je tedy $\mathsf{Post}(h_P) = \{M \mid M \subseteq V\}$. Největší pevný bod je supremum množiny všech postfixních bodů, tj. $\nu h_P = \bigcup \{M \mid M \subseteq V\}$, protože suprema jsou množinová sjednocení. Protože $\bigcup \{M \mid M \subseteq V\} = V$, je množina $V$ největším pevným bodem zobrazení $h_P$.

4. Extense boxu — zobrazení $\| [R] \|_{\mathbb{V}}$. Označme toto zobrazení jako $r$.

(a) Nejmenší pevný bod. Množina $M \subseteq V$ je prefixním bodem zobrazení $r$ právě tehdy, když platí $r(M) \subseteq M$. To nastává právě tehdy, když je splněna podmínka:

([$R$]-B) Je-li vrchol $v$ takový, že pro libovolnou akci $v \xrightarrow{R} v'$ je $v' \in M$, potom $v \in M$.

Množina vrcholů $M$ je tedy prefixním bodem zobrazení $r$ přesně tehdy, když množina $M$ je uzavřena na akci $R$ „směrem dozadu“.

Nejmenší pevný bod je infimum množiny všech prefixních bodů, tj. $\mu r$ je nejmenší množina vrcholů s vlastností ([$R$]-B).

(b) Největší pevný bod. Množina $M \subseteq V$ je postfixním bodem zobrazení $r$ právě tehdy, když platí $M \subseteq r(M)$. To nastává právě tehdy, když je splněna podmínka:

([$R$]-F) Je-li vrchol $v$ v množině $M$, pak pro libovolnou akci $v \xrightarrow{R} v'$ je $v' \in M$.

Množina vrcholů $M$ je tedy postfixním bodem zobrazení $r$ přesně tehdy, když množina $M$ je uzavřena na akci $R$ „směrem dopředu“.

Největší pevný bod je supremum množiny všech postfixních bodů, tj. $\nu r$ je největší množina vrcholů s vlastností ([$R$]-F). Množina všech vrcholů $V$ je však evidentně postfixním bodem a větší takový nelze nalézt, tj. platí $\nu r = V$.

Dosavadní výsledky o pevných bodech nám umožní ukázat, že funkce $f$ pro definici nejslabší liberální prepodmínky `w`*while* cyklu je monotónní a tudíž existuje její největší pevný bod. Podle druhého tvrzení lemmatu 8.2.1 je tento největší pevný bod extensí nejslabší liberální prepodmínky.

**8.3.8 Příklad** Připomeňme, že pro cyklus `while` $b$ `do` $q$ `od` a formuli $\varphi$ bez modalit jsme definovali funkci $f : exp(V) \to exp(V)$ předpisem:

$$f : X \mapsto \Big( \| [q] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}} \Big) \cup \Big( \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}} \Big)$$

kde $\mathscr{M}$ je zamýšlený modální model jazyka `w`*while* programů. Označme $B = \| b \|_{\mathscr{M}}$ a $P = \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$. Povšimněme si, že množiny $B$ a $P$ na množině $X$ nezávisí — jsou to konstantní množiny pamětí.

Funkci $f$ lze tedy psát jako $f : X \mapsto (\| [q] \|_{\mathscr{M}}(X) \cap B) \cup P$ a tato funkce je zřejmě monotónní, vznikla totiž jako složení tří monotónních funkcí (ve značení příkladu 8.3.3):

$$r : X \mapsto \| [q] \|_{\mathscr{M}}(X) \qquad g_B : X \mapsto X \cap B \quad \text{a} \quad h_M : X \mapsto X \cup P$$

Podle Tarského věty má tedy funkce $f$ nejmenší pevný bod $\mu f$ a největší pevný bod $\nu f$.

Podle druhého tvrzení lemmatu 8.2.1 je největší pevný bod zobrazení $f$ extensí nejslabší liberální prepodmínky $\mathsf{wlp}(\varphi \mid \texttt{while}\ b\ \texttt{do}\ q\ \texttt{od})$.

## 8.4 Dualita pevných bodů

V předchozím odstavci jsme naznačili, že Tarského větu o pevných bodech budeme používat pro monotónní zobrazení $f : exp(V) \to exp(V)$. Množina $exp(V)$ je jistě úplný svaz, má však ještě strukturu Booelovy algebry: výsledek operace komplementu k množině $X \in exp(V)$ (tj. $X \subseteq V$) je množinový doplněk, tj. množina $V \setminus X$.

V případě, že studujeme monotónní zobrazení na úplném svazu, který je navíc Booleovou algebrou, stačí umět počítat buď nejmenší pevné body (největší pevný bod je potom nejmenším pevným bodem jistého *duálního* zobrazení), nebo stačí umět počítat největší pevné body (nejmenší pevný bod je potom největším pevným bodem jistého *duálního* zobrazení).

**8.4.1 Definice** *Úplná Booleova algebra* je Booleova algebra $\langle X, \sqcap, \sqcup, {}^{-1}, \top, \bot \rangle$ taková, že svaz $\langle X, \sqcap, \sqcup \rangle$ je úplný.

Příkladem úplné Booleovy algebry je tedy svaz $\langle exp(V), \subseteq \rangle$. I když v aplikacích nám půjde právě o tuto úplnou Booleovu algebru, následující tvrzení zformulujeme obecně. Připomeňme, že podle cvičení 7.7.3 můžeme volně zaměňovat pojem (abstraktního) svazu a pojem posetu, kde každá dvojice prvků má supremum a infimum. Zápisem $x \sqsubseteq y$ tedy rozumíme platnost rovnosti $x \sqcap y = x$ (nebo, ekvivalentně, platnost rovnosti $x \sqcup y = y$).

**8.4.2 Tvrzení** *Ať $\mathbb{X} = \langle X, \sqcap, \sqcup, {}^{-1}, \top, \bot \rangle$ je úplná Booleova algebra. Potom platí:*

1. *Pro libovolná $x$, $y$ v $X$ platí $x \sqsubseteq y$ právě tehdy, když platí $y^{-1} \sqsubseteq x^{-1}$.*
2. *Ať $f : \mathbb{X} \to \mathbb{X}$ je monotónní zobrazení. Jako $\tilde{f} : \mathbb{X} \to \mathbb{X}$ označte zobrazení definované rovností*

$$\tilde{f}(x) = f(x^{-1})^{-1}.$$

*Potom $\tilde{f}$ je monotónní zobrazení a platí*

$$\mu f = (\nu \tilde{f})^{-1} \quad \text{a} \quad \nu f = (\mu \tilde{f})^{-1}.$$

Důkaz. 1. Nerovnost $x \sqsubseteq y$ platí právě tehdy, když platí rovnost $x \sqcap y = x$. Aplikací komplementu tedy dostaneme platnost rovnosti $(x \sqcap y)^{-1} = x^{-1}$. Na levou stranu této rovnosti aplikujme De Morganovo pravidlo (jedná se o jeden z axiomů pro Booleovu algebru) a dostaneme rovnost $x^{-1} \sqcup y^{-1} = x^{-1}$. Poslední rovnost znamená přesně to, že platí nerovnost $y^{-1} \sqsubseteq x^{-1}$.

Obrácená implikace se dokáže analogicky.

2. Nejprve ukážeme, že zobrazení $\tilde{f}$ je monotónní. Vezměme $x$, $y$ tak, že $x \sqsubseteq y$. Potom podle tvrzení 1. platí nerovnost $y^{-1} \sqsubseteq x^{-1}$. Protože $f$ je monotónní, dostáváme $f(y^{-1}) \sqsubseteq f(x^{-1})$. Aplikací tvrzení 1. na poslední nerovnost dostáváme nerovnost $f(x^{-1})^{-1} \sqsubseteq f(y^{-1})^{-1}$, což je — podle definice zobrazení $\tilde{f}$ — nerovnost $\tilde{f}(x) \sqsubseteq \tilde{f}(y)$.

Zobrazení $\tilde{f}$ tedy má podle Tarského věty největší pevný bod $\nu \tilde{f}$. Označme jako $m$ jeho komplement. Chceme ukázat, že $m$ je nejmenší pevný bod zobrazení $f$.

Zřejmě platí $f(m)^{-1} = \tilde{f}(\nu \tilde{f}) = \nu \tilde{f} = m^{-1}$, a tudíž $m$ je pevným bodem zobrazení $f$.

Předpokládejme, že $x$ je nějaký pevný bod zobrazení $f$. Chceme ukázat, že potom platí nerovnost $m \sqsubseteq x$. Především platí, že $x^{-1}$ je pevným bodem zobrazení $\tilde{f}$, protože $\tilde{f}(x^{-1}) = f(x)^{-1} = x^{-1}$. Tudíž víme, že platí nerovnost $x^{-1} \sqsubseteq \nu \tilde{f}$, protože $\nu \tilde{f}$ je největší pevný bod zobrazení $\tilde{f}$. Z nerovnosti $x^{-1} \sqsubseteq \nu \tilde{f}$ ale plyne, že $(\nu \tilde{f})^{-1} \sqsubseteq x$. Protože $(\nu \tilde{f})^{-1} = m$, je důkaz hotov.

Rovnost $\nu f = (\mu \tilde{f})^{-1}$ se dokáže analogicky. ■

**8.4.3 Definice** Zobrazení $\tilde{f}$ z předchozího tvrzení nazýváme *duální zobrazení k zobrazení* $f$.

Tvrzení 8.4.2 nyní budeme aplikovat na monotónní zobrazení z příkladu 8.3.7.

**8.4.4 Příklad** V příkladu 8.3.7 jsme spočítali největší a nejmenší body následujících monotónních zobrazení. Nyní ke každému zobrazení spočteme jeho duální zobrazení a potom aplikujeme tvrzení 8.4.2.

1. Konstantní zobrazení $f_P$. Toto zobrazení má jako duální zobrazení konstantní zobrazení na množinu $V \setminus P$. Platí totiž $\tilde{f_P}(X) = V \setminus f_P(V \setminus X) = V \setminus P$.

   Největší pevný bod $\tilde{f_P}$ je samozřejmě množina $V \setminus P$. Komplement k $V \setminus P$ dává nejmenší pevný bod zobrazení $f_P$: jde o množinu $V \setminus (V \setminus P) = P$.

   Nejmenší pevný bod $\tilde{f_P}$ je také množina $V \setminus P$. Komplement k $V \setminus P$ dává největší pevný bod zobrazení $f_P$: jde o množinu $V \setminus (V \setminus P) = P$.

2. Průnik s pevnou množinou — zobrazení $g_P$. Toto zobrazení má jako duální zobrazení sjednocení s množinou $V \setminus P$. Platí totiž $\tilde{g_P}(X) = V \setminus g_P(V \setminus X) = V \setminus ((V \setminus X) \cap P) = X \cup (V \setminus P)$.

   Největší pevný bod $\tilde{g_P}$ je množina $V$. Komplement k $V$ dává nejmenší pevný bod zobrazení $g_P$: jde o prázdnou množinu.

   Nejmenší pevný bod $\tilde{g_P}$ je množina $V \setminus P$. Komplement k $V \setminus P$ dává největší pevný bod zobrazení $g_P$: jde o množinu $P$.

3. Sjednocení s pevnou množinou — zobrazení $h_P$. Toto zobrazení má jako duální zobrazení průnik s množinou $V \setminus P$. Platí totiž $\tilde{h_P}(X) = V \setminus h_P(V \setminus X) = V \setminus ((V \setminus X) \cup P) = X \cap (V \setminus P)$.

   Největší pevný bod $\tilde{h_P}$ je množina $V \setminus P$. Komplement k $V \setminus P$ dává nejmenší pevný bod zobrazení $g_P$: jde o množinu $P$.

   Nejmenší pevný bod $\tilde{h_P}$ je množina $\emptyset$. Komplement k $\emptyset$ dává největší pevný bod zobrazení $g_P$: jde o množinu $V$.

4. Předpokládejme, že $V$ je množina vrcholů akční struktury $\mathbb{V}$ a že $R$ je akce. Extensi boxu — zobrazení $\| [R] \|_{\mathbb{V}}$ — jsme označili jako $r$.

   Duální zobrazení k $r$ je extense diamantu $R$: to plyne okamžitě z definice diamantu jako syntaktické zkratky.

## 8.5 Motivace syntaxe a sémantiky $\mu$-modálních formulí

Připomeňme, že v definici Hennessyho-Milnerovy modální logiky z odstavce 4.1 jsme definovali modální logiku nad množinou atomických formulí $At$. Tato množina smí být prázdná — volbu množiny $At$ si můžeme představit jako volbu vyjadřovací síly jazyka, kterým se budeme vyjadřovat. Jedná se tedy o parametr, jehož volba závisí na nás. Pro definici formulí modálních $\mu$-formulí budeme však potřebovat parametr další — množinu proměnných $Var$. Budeme předpokládat, že množina $Var$ je dána jednou provždy: je to spočetná množina, jejíž typický prvek budeme značit $x$.

Formule modálního $\mu$-kalkulu budeme opět interpretovat v akčních strukturách typu $\Delta$. Každé formuli $\varphi$ přiřadíme její extensi — množinu vrcholů, ve kterých je formule $\varphi$ vynucena v modelu $\mathscr{M}$. Jak uvidíme, formule budou obsahovat proměnné: extensi takové formule si potom představíme jako funkci, která množině vrcholů přiřadí množinu vrcholů. Tato představa způsobí několik potíží, na které v tomto odstavci upozorníme. Přesnou definici syntaxe a sémantiky modálního $\mu$-kalkulu podáme v definicích 8.5.10 a 8.5.11.

Začneme motivačním příkladem.

**8.5.1 Příklad** Připomeňme, že v příkladu 8.3.8 jsme ukázali, že extense nejslabší liberální prepodmínky cyklu `while` $b$ `do` $q$ `od` vzhledem k formuli $\varphi$ je největším pevným bodem funkce $f : exp(V) \to exp(V)$ definované předpisem:

$$f : X \mapsto \Big( \| [q] \|_{\mathscr{M}}(X) \cap B \Big) \cup P$$

kde $B = \| b \|_{\mathscr{M}}$ a $P = \| \varphi \|_{\mathscr{M}} \cap \| \neg b \|_{\mathscr{M}}$.

Zatím nic nezaručuje, že největší pevný bod $f$ je skutečně extensí nějaké formule. Tuto obtíž odstraníme tím, že v modálním $\mu$-kalkulu budeme mít zaručeno, že největší pevný bod monotónního zobrazení je vždy extensí formule.

Formuli $\mathsf{wlp}(\varphi \mid \texttt{while}\ b\ \texttt{do}\ q\ \texttt{od})$ v modálním $\mu$-kalkulu označíme jako $\nu x.\alpha$. Toto značení volíme proto, že pak formule $\alpha$ intuitivně odpovídá funkci $f$. Ve formuli $\alpha$ by se tedy měl odrazit fakt, že funkce $f$ má jeden argument. Toho docílíme tím, že připustíme, aby se ve formuli $\alpha$ vyskytovala proměnná $x$.

Tvar formule $\alpha$ bude $([q]x \wedge b) \vee (\varphi \wedge \neg b)$. Překlad formule $\alpha$ na zobrazení $f$ pak proběhne zřejmým způsobem.

Z předchozího příkladu je patrné, že budeme muset zaručit korespondenci formulí obsahujících proměnné s monotónními zobrazeními. Učiníme nyní několik pokusů o definici formulí modálního $\mu$-kalkulu. Tyto definice budeme postupně vylepšovat a výsledkem bude přesná definice 8.5.10.

**8.5.2 První pokus o syntaxi modálního $\mu$-kalkulu** První pokus bude neúspěšný proto, že si nebudeme vědět rady, jak formuli modálního $\mu$-kalkulu interpretovat. Formule $\mu$-kalkulu je řetězec $\varphi$ vytvořený podle následujících pravidel:

1. Každá atomická formule $a$ je formulí modálního $\mu$-kalkulu.

   Této formuli v modálním $\mu$-modelu přiřadíme množinu vrcholů akční struktury, ve kterých je $a$ pravdivá.

2. Každá proměnná $x$ je formulí modálního $\mu$-kalkulu.

   Proměnné $x$ v modálním $\mu$-modelu přiřadíme *identické zobrazení* $1_{exp(V)} : exp(V) \to exp(V)$ jako její extensi. Intuitivně takto skutečně musíme proměnnou interpretovat, chceme-li dodržet korespondenci z příkladu 8.5.1 mezi formulemi, které obsahují proměnné, a monotónními zobrazeními.

3. Formule $\texttt{tt}$ je formulí modálního $\mu$-kalkulu.

   Této formuli v modálním $\mu$-modelu přiřadíme množinu $V$ všech vrcholů akční struktury.

4. Jsou-li $\varphi_1$ a $\varphi_2$ formule modálního $\mu$-kalkulu, potom $\varphi_1 \wedge \varphi_2$ je formule modálního $\mu$-kalkulu.

   V modálním $\mu$-modelu bychom měli formuli $\varphi_1 \wedge \varphi_2$ přiřadit průnik extensí formulí $\varphi_1$ a $\varphi_2$. Co však dělat, když $\varphi_1$ je proměnná $x$ a $\varphi_2$ je formule $\texttt{tt}$? Co rozumíme průnikem funkce s množinou?

Dále již nemá smysl pokračovat. Myšlenka, že modální $\mu$-model bude pouhým přenesením definice modálního modelu, není správná.

Před uvedením druhého pokusu o syntaxi a sémantiku formulí modálního $\mu$-kalkulu připomeneme, jakým způsobem se s přítomností proměnných vyrovnává sémantika formulí jazyka predikátové logiky. Pokud čtenář není s predikátovou logikou obeznámen, ať tuto poznámku přeskočí.

**8.5.3 Poznámka** Z definice sémantiky formulí predikátové logiky je dobře znám pojem *kontextu pro proměnné*.

Připomeňme, že jazyk $\mathcal{L}$ predikátové logiky je dán standardní spočetnou množinou proměnných $Var$, neprázdnou množinou predikátových $\mathcal{P}$, množinou funkčních symbolů $\mathcal{F}$ a množinou konstantních symbolů $\mathcal{K}$. Množina $Var$ je dána jednou provždy, u predikátových a funkčních symbolů a konstant máme volnost; jejich volba určuje vyjadřovací sílu jazyka $\mathcal{L}$. U každého predikátového a funkčního symbolu musí být dále specifikována jeho arita — „počet argumentů“. Tyto deklarace umožňují vytvoření množiny všech formulí jazyka $\mathcal{L}$.

Uvedeme jednoduchý příklad: jazyk $\mathcal{L}$ je dán následovně:

- Množina predikátových symbolů $\mathcal{P}$ obsahuje jediný prvek: symbol $S$. Arita symbolu $S$ je rovna číslu 1.
- Množina funkčních symbolů $\mathcal{F}$ obsahuje jediný prvek: symbol $f$. Arita symbolu $S$ je rovna číslu 2.
- Množina konstant $\mathcal{K}$ obsahuje jediný prvek: symbol $a$.

Víme-li, že $x \in Var$, pak výše uvedená specifikace nám dovolí prohlásit, že následující řetězce znaků jsou formulemi jazyka $\mathcal{L}$:

$$S(a) \qquad S(f(x,a)) \qquad \forall x\, S(f(x,a))$$

Samozřejmě, tyto tři řetězce „nic neznamenají“ — symboly $S$, $f$ a $a$ nejsou nijak interpretovány. Avšak samotná interpretace těchto symbolů ještě nezaručí, že každá z výše uvedených formulí se interpretuje buď jako pravdivá nebo jako nepravdivá.

Uvedeme příklad interpretace symbolů $S$, $f$ a $a$:

- Interpretovat budeme na množině všech přirozených čísel $\mathbb{N}$. Tím jsme se rozhodli pro „datový typ“ — přirozená čísla.

  Interpretace predikátového symbolu $S$ musí být nějaký „Booleovský predikát“ nad přirozenými čísly, který vyžaduje jeden argument (aritu symbolu $S$ jsme deklarovali jako číslo 1).

  Podobně, interpretace funkčního symbolu $f$ musí být nějaký „operátor“ nad přirozenými čísly, který vyžaduje dva argumenty (aritu symbolu $f$ jsme deklarovali jako číslo 2).

  Konečně interpetace konstantního symbolu $a$ musí být nějaké pevné přirozené číslo.

- Symbol $S$ interpretujeme jako množinu sudých čísel. Přesněji: položíme

$$I(S) = \{n \in \mathbb{N} \mid n \text{ je sudé číslo } \}.$$

- Symbol $f$ interpretujeme jako sčítání. Přesněji: položíme $I(f) : (n, m) \mapsto n + m$.

- Symbol $a$ interpretujeme jako číslo 11, přesněji, položíme $I(a) = 11$.

V této interpretaci se pak řetězec $S(a)$ přeloží jako tvrzení: *Přirozené číslo* 11 *je sudé.* To je nepravdivé tvrzení.

Řetězec $\forall x\, S(f(x, a))$ se intuitivně přeloží jako tvrzení: *Pro každé přirozené číslo* $n$ *platí:* $n + 11$ *je sudé číslo.* I toto tvrzení je nepravdivé. Pokud však sémantika formulí predikátové logiky je řízena syntaxí (a je tomu tak u všech formálních jazyků), měli bychom před porozuměním řetězci $\forall x\, S(f(x, a))$ nejprve rozumět syntakticky jednoduššímu řetězci $S(f(x, a))$. Tento řetězec se překládá jako tvrzení $n + 11$ *je sudé číslo.* Toto tvrzení může být pravdivé (například pro $n$ rovno 127), může však být i nepravdivé (například pro $n$ rovno 156). Zjišťujeme, že sémantika řetězce $S(f(x, a))$ není jasná. Nevíme totiž vůbec nic o interpretaci proměnné $x$: bude mít „hodnotu“ 127 nebo 156?

Tento problém lze obejít *interpretací v kontextu.* Nestačí tedy interpretovat pouze symboly jazyka, musíme ještě interpretovat každou proměnnou (tj. zadat její „aktuální hodnotu“). Této interpretaci proměnných říkáme *kontext pro proměnné.* Přesněji: kontextem pro proměnné rozumíme (v našem speciálním případě) funkci $\rho$ : $Var \to \mathbb{N}$. Znamená to, že o každé proměnné víme, jakou má aktuální hodnotu.

Teprve znalost kontextu dovolí dodržet dogma o tom, že *sémantika je řízená syntaxí* neboli, že k pochopení složitějšího řetězce se dostaneme pochopením jednodušších řetězců.

Předvedeme nyní, jak interpretovat $S(f(x, a))$ a $\forall x\, S(f(x, a))$ v kontextu $\rho$, kde $\rho(x) = 127$.

Formule $S(f(x, a))$ je interpretována jako pravdivé tvrzení *Číslo* $127 + 11$ *je sudé.*

U formule $\forall x\, S(f(x, a))$ je situace o něco složitější: musíme ukázat, že $S(f(x, a))$ je interpretována jako pravdivá v každém kontextu $\rho'$, kde kontexty $\rho$ a $\rho'$ se liší pouze hodnotou v proměnné $x$. Snadno však nalezneme takový kontext $\rho'$, ve kterém je $S(f(x, a))$ interpretována jako nepravdivá: jde například o kontext

$$\rho'(t) = \begin{cases} 156, & \text{je-li } t \text{ proměnná } x \\ \rho(t), & \text{je-li proměnná } t \text{ různá od } x \end{cases}$$

Ukázali jsme, že v původním kontextu $\rho$ je formule $\forall x\, S(f(x, a))$ interpretována jako nepravdivá.

Samozřejmě, výše uvedený postup lze použít pro libovolný počáteční kontext $\rho$. Ukázali jsme tedy, že formule $\forall x\, S(f(x, a))$ je interpretována jako nepravdivá v libovolném kontextu.

**8.5.4 Druhý pokus o syntaxi modálního $\mu$-kalkulu** Formule budeme interpretovat jako množiny vrcholů v akční struktuře a budeme vždy interpretovat v takovém kontextu, kdy bude jasné, jakou hodnotu mají jednotlivé proměnné. Přesněji: *kontextem pro proměnné* budeme rozumět funkci $\rho$ : $Var \to exp(V)$, která proměnné $x$ přiřadí množinu $\rho(x)$ těch vrcholů, ve kterých je proměnná $x$ pravdivá. Každé formuli $\varphi$ pak přiřadíme její extensi $\| \varphi \|_{\mathscr{M}}, \rho$ v závislosti na kontextu $\rho$.

U každé formule $\varphi$ modálního $\mu$-kalkulu budeme muset určit množinu $\mathsf{BV}(\varphi)$ vázaných proměnných a množinu $\mathsf{FV}(\varphi)$ volných proměnných.

Formule $\mu$-kalkulu je řetězec $\varphi$ vytvořený podle následujících pravidel:

1. Každá atomická formule $a$ je formulí modálního $\mu$-kalkulu. Tato formule neobsahuje žádnou proměnnou, volnou ani vázanou: $\mathsf{FV}(a) = \mathsf{BV}(a) = \emptyset$.

   Této formuli v modálním $\mu$-modelu přiřadíme množinu vrcholů akční struktury, ve kterých je $a$ pravdivá $\| a \|_{\mathscr{M},\rho}$. Tato množina není na kontextu proměnných $\rho$ závislá.

2. Každá proměnná $x$ je formulí modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(x) = \emptyset$ a $\mathsf{FV}(x) = \{x\}$.

   V kontextu $\rho$ má proměnná $x$ extensi $\| x \|_{\mathscr{M},\rho} = \rho(x)$.

3. Formule $\mathtt{tt}$ je formulí modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(\mathtt{tt}) = \mathsf{FV}(\mathtt{tt}) = \emptyset$.

   Této formuli v modálním $\mu$-modelu přiřadíme množinu $V$ všech vrcholů akční struktury $\| \mathtt{tt} \|_{\mathscr{M},\rho} = V$. Tato množina není na kontextu proměnných $\rho$ závislá.

4. Jsou-li $\varphi_1$ a $\varphi_2$ formule modálního $\mu$-kalkulu, potom $\varphi_1 \wedge \varphi_2$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(\varphi_1 \wedge \varphi_2) = \mathsf{BV}(\varphi_1) \cup \mathsf{BV}(\varphi_2)$ a $\mathsf{FV}(\varphi_1 \wedge \varphi_2) = \mathsf{FV}(\varphi_1) \cup \mathsf{FV}(\varphi_2)$.

   V kontextu $\rho$ přiřadíme formuli $\varphi_1 \wedge \varphi_2$ jako extensi průnik extensí formulí $\varphi_1$ a $\varphi_2$ v kontextu $\rho$: $\| \varphi_1 \wedge \varphi_2 \|_{\mathscr{M},\rho} = \| \varphi_1 \|_{\mathscr{M},\rho} \cap \| \varphi_2 \|_{\mathscr{M},\rho}$.

5. Jestliže $\varphi$ je formule modálního $\mu$-kalkulu, potom $\neg\varphi$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(\neg\varphi) = \mathsf{BV}(\varphi)$ a $\mathsf{FV}(\neg\varphi) = \mathsf{FV}(\varphi)$.

   V kontextu $\rho$ přiřadíme formuli $\neg\varphi$ jako extensi doplněk extense formule $\varphi$ v kontextu $\rho$: $\| \neg\varphi \|_{\mathscr{M},\rho} = V \setminus \| \varphi \|_{\mathscr{M},\rho}$.

6. Jestliže $\varphi$ je formule modálního $\mu$-kalkulu a $R$ je jméno akce, potom $[R]\varphi$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}([R]\varphi) = \mathsf{BV}(\varphi)$ a $\mathsf{FV}([R]\varphi) = \mathsf{FV}(\varphi)$.

   V kontextu $\rho$ přiřadíme formuli $[R]\varphi$ jako extensi hodnotu extense formule $\varphi$ v kontextu $\rho$ při zobrazení $\| [R] \|_{\mathscr{M}}$: $\| [R]\varphi \|_{\mathscr{M},\rho} = \| [R] \|_{\mathscr{M}}(\| \varphi \|_{\mathscr{M},\rho})$.

7. Je-li $\varphi$ formule modálního $\mu$-kalkulu a je-li $x$ proměnná, potom $\nu x.\varphi$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(\nu x.\varphi) = \mathsf{BV}(\varphi) \cup \{x\}$ a $\mathsf{FV}(\nu x.\varphi) = \mathsf{FV}(\varphi) \setminus \{x\}$. Prefix $\nu x.$ se tedy chová podobně jako kvantifikátory v predikátové logice — prefix $\nu x.$ *váže* výskyty proměnné $x$ ve formuli $\varphi$.

   V kontextu $\rho$ přiřadíme formuli $\nu x.\varphi$ jako extensi *největší pevný bod funkce $f_\varphi$ jedné proměnné*, a to takto:

   (a) Je-li proměnná $x$ vázaná ve formuli $\varphi$, potom funkce $f_\varphi$ bude konstantní funkce s hodnotou extense formule $\varphi$: $f_\varphi : X \mapsto \| \varphi \|_{\mathscr{M},\rho}$.

   (b) Je-li proměnná $x$ volná ve formuli $\varphi$, potom extense $\varphi$ závisí na kontextu proměnných $\rho$. Toho využijeme.

   Funkce $f_\varphi$ přiřadí množině vrcholů $X$ množinu $f_\varphi(X)$ definovanou takto: nejprve změníme kontext proměnných $\rho$ na kontext $\rho'$ tak, aby v kontextu $\rho'$ měla proměnná hodnotu $X$. Ostatní proměnné mají v kontextu $\rho'$ nezměněnou hodnotu.

   Hodnotu funkce $f_\varphi$ v $X$ definujeme jako množinu $\| \varphi \|_{\mathscr{M},\rho'}$.

Spojky $\vee$, $\Rightarrow$, $\Leftrightarrow$, modality $\langle R \rangle$ a formuli $\mathtt{ff}$ zavádíme obvyklým způsobem jako syntaktické zkratky. Formuli $\mu x.\varphi$ zavedeme jako syntaktickou zkratku za formuli $\neg\nu x.(\neg\varphi[x \mapsto \neg x])$, kde jako $\varphi[x \mapsto \neg x]$ označujeme formuli vzniklou z formule $\varphi$ náhradou každého *volného* výskytu proměnné $x$ formulí $\neg x$ — viz tvrzení 8.4.2.

Než v příkladu 8.5.6 vysvětlíme, proč ani definice 8.5.4 syntaxe a sémantiky modálních $\mu$-formulí není uspokojivá, přiblížíme na příkladu technickou definici extense formule $\nu x.\varphi$. Využijeme k tomu opět příkladu nejslabší liberální prepodmínky $\mathtt{while}$ cyklu.

**8.5.5 Příklad** V příkladu 8.5.1 jsme jako $\alpha$ označili formuli $([q]x \wedge b) \vee (\varphi \wedge \neg b)$ a jako $\nu x.\alpha$ nejslabší liberální prepodmínku $\mathsf{wlp}(\varphi \mid \mathtt{while}\ b\ \mathtt{do}\ q\ \mathtt{od})$. Podle předchozí definice jsou $\alpha$ i $\nu x.\alpha$ zřejmě modální $\mu$-formule.

Kontext proměnných $\rho$ je funkce, která každé proměnné $x$ přiřadí množinu pamětí $\rho(x)$. Zvolme takový kontext $\rho$ pevně a spočítáme extensi jednotlivých formulí. Extense atomických formulí zvolíme stejně jako v zamýšleném modálním modelu. To je v pořádku: podle definice extense atomických formulí na kontextu nezávisejí.

1. Proměnná $x$ má v kontextu $\rho$ extensi $\| x \|_{\mathscr{M},\rho} = \rho x$.

2. Formule $\varphi \wedge \neg b$ má extensi na kontextu $\rho$ nezávislou, protože ani $\varphi$ ani $b$ neobsahují proměnnou modálního $\mu$-kalkulu. (Samozřejmě: ve formulích $\varphi$ a $b$ se mohou vyskytovat proměnné pro identifikátory jazyka $\mathtt{while}$ programů — to však nejsou proměnné modálního $\mu$-kalkulu!)

   $\| \varphi \wedge \neg b \|_{\mathscr{M},\rho}$ je tedy množina pamětí, ve kterých je vynucena formule $\varphi \wedge \neg b$.

3. Formule $[R]x$ má v kontextu $\rho$ extensi $\| [R]x \|_{\mathscr{M},\rho}$ rovnu funkční hodnotě funkce $\| [R] \|_{\mathscr{M}}$ v bodě $(\| x \|_{\mathscr{M},\rho})$, což je množina $\| [R] \|_{\mathscr{M}}(\rho(x))$.

4. Extense formule $\alpha$ v kontextu $\rho$ je tedy množina

$$\Big( \| [R] \|_{\mathscr{M}}(\rho(x)) \cap \| b \|_{\mathscr{M}} \Big) \cup \| \varphi \wedge \neg b \|_{\mathscr{M}}.$$

   Tato množina je ovšem na kontextu závislá následujícím způsobem: změníme-li kontext $\rho$ na kontext $\rho'$ tak, že proměnná $x$ má v kontextu $\rho'$ hodnotu $X$, potom je extense formule $\alpha$ v kontextu $\rho'$ rovna množině

$$\Big( \| [R] \|_{\mathscr{M}}(X) \cap \| b \|_{\mathscr{M}} \Big) \cup \| \varphi \wedge \neg b \|_{\mathscr{M}}.$$

   Zjišťujeme tedy, že funkce $f_\alpha$ z definice extense formule $\nu x.\alpha$ není nic jiného, než funkce $f$ z příkladu 8.3.8.

5. Extense formule $\nu x.\alpha$ je největší pevný bod zobrazení $f_\alpha$. Podle druhého tvrzení lemmatu 8.2.1 je největší pevný bod zobrazení $f_\alpha$ extensí nejslabší liberální prepodmínky $\mathtt{wlp}(\varphi \mid \mathtt{while}\ b\ \mathtt{do}\ q\ \mathtt{od})$.

V následujícím příkladu osvětlíme, proč ani druhý pokus o definici syntaxe a sémantiky modálního $\mu$-kalkulu nebyl úspěšný — existují totiž formule, kterým neumíme přiřadit význam.

**8.5.6 Příklad** Formule $\neg x$, kde $x$ je proměnná, je podle pravidel z 8.5.4 formule modálního $\mu$-kalkulu. Její extense v kontextu $\rho$ je množina $V \setminus \rho(x)$.

Formule $\nu x.\neg x$ je také formule modálního $\mu$-kalkulu — označme ji $\omega$. Její extense v kontextu $\rho$ je největší pevný bod jistého zobrazení $f_\omega : exp(V) \to exp(V)$.

Jak zobrazení $f_\omega$ pracuje? Zvolme $X \subseteq V$ a změňme kontext $\rho$ na kontext $\rho'$, ve kterém má proměnná $x$ hodnotu $X$. Hodnota $f_\omega(X)$ je extense formule $\neg x$ v kontextu $\rho'$. Tato extense je ovšem množina $V \setminus X$.

Zobrazení $f_\omega$ je tedy zobrazením doplňku na $exp(V)$. Z příkladu 8.3.3 víme, že toto zobrazení není monotónní. Navíc zřejmě neexistuje vůbec žádný pevný bod tohoto zobrazení. Formuli $\nu x.\neg x$ tedy podle pravidel 8.5.4 nelze přiřadit žádnou extensi.

Při definici syntaxe modálních $\mu$-formulí tvaru $\nu x.\varphi$ se tedy budeme muset omezit na takové formule $\varphi$, pro něž je zobrazení $f_\varphi$ monotónní. Vymezení takových formulí musí mít syntaktický charakter a je obsahem následující definice. Tato definice využívá pojmu syntaktického stromu pro formule vytvořené podle pravidel uvedených v 8.5.4. Přesnou definici syntaktických stromů těchto formulí přenecháváme jako cvičení 8.8.4.

**8.5.7 Definice** Řekneme, že volná proměnná $x$ má ve formuli $\varphi$ vytvořené podle pravidel uvedených v 8.5.4 *positivní výskyt*, pokud na každé cestě od listu označeném proměnnou $x$ ke kořeni syntaktického stromu formule $\varphi$ leží sudý počet negací.

**8.5.8 Příklad** V následujících formulích má proměnná $x$ positivní výskyt: $x$, $[R]x$, $x \wedge b$, $x \vee x$, $\nu y.(x \vee b)$.

Formule $x$ nemá positivní výskyt například ve formuli $x \Rightarrow x$ nebo $x \wedge \neg x$.

Smyslem dalšího lemmatu je ukázat, že pokud se omezíme na formule s positivními výskyty proměnných, pak v pokusu 8.5.4 o definici modálního $\mu$-kalkulu nenarazíme na žádné obtíže.

**8.5.9 Lemma** *Jestliže má proměnná $x$ positivní výskyt ve formuli $\varphi$, potom je zobrazení $f_\varphi$ definované v bodě 7 pravidel 8.5.4 monotónní.*

Důkaz. Tvrzení je triviální, pokud se ve formuli $\varphi$ proměnná $x$ nevyskytuje vůbec nebo jen jako vázaná proměnná.

V případě, že se proměnná $x$ ve formuli $\varphi$ vyskytuje volně, budeme postupovat následovně.

1. Nejprve formuli $\varphi$ přepíšeme do positivní formy, tj. tak, aby se v ní vyskytovaly pouze spojky disjunkce, konjunkce, boxy, diamanty, prefixy $\nu$ a $\mu$ a aby spojka negace se vyskytovala pouze těsně před atomickými formulemi a proměnnými. To jde vždy udělat — viz cvičení 8.8.5. Takto přepsanou formuli $\varphi$ označme $\mathrm{p}(\varphi)$. Lze ukázat, že extense $\varphi$ a $\mathrm{p}(\varphi)$ jsou totožné množiny (v libovolném modelu a libovolném kontextu).

2. Pokud již máme positivní formu formule $\varphi$, potom podle předpokladů v jejím syntaktickém stromu při cestě od listu označeném $x$ ke kořeni nenarazíme na žádnou spojku negace.

3. Nyní zbývá postupovat indukcí podle syntaktické složitosti positivní formy formule $\varphi$. Uvedeme jen některé případy, pro zbytek odkazujeme na cvičení 8.8.6.

   (a) Formule $\mathrm{p}(\varphi)$ je proměnná $x$. Platí $f_{\mathrm{p}(\varphi)}(X) = X$ a to je monotónní zobrazení.

   (b) Formule $\mathrm{p}(\varphi)$ je tvaru $\varphi_1 \wedge \varphi_2$. Protože platí $f_{\mathrm{p}(\varphi)}(X) = f_{\varphi_1}(X) \cap f_{\varphi_2}(X)$, je $f_{\mathrm{p}(\varphi)}$ monotónní zobrazení.
   Je-li $X_1 \subseteq X_2$, potom platí $f_{\varphi_1}(X_1) \cap f_{\varphi_2}(X_1) \subseteq f_{\varphi_1}(X_2) \cap f_{\varphi_2}(X_2)$. Předpokládáme totiž, že zobrazení $f_{\varphi_1}$ a $f_{\varphi_2}$ jsou monotónní.

   (c) Formule $\mathrm{p}(\varphi)$ je tvaru $\varphi_1 \vee \varphi_2$ — viz cvičení 8.8.6.

   (d) Formule $\mathrm{p}(\varphi)$ je tvaru $[R]\varphi_1$. Protože platí $f_{\mathrm{p}(\varphi)}(X) = \| [R] \|_{\mathscr{M},\rho}(f_{\varphi_1}(X))$, je $f_{\mathrm{p}(\varphi)}$ monotónní zobrazení.

   (e) Formule $\mathrm{p}(\varphi)$ je tvaru $\langle R\rangle\varphi_1$ — viz cvičení 8.8.6.

   (f) Formule $\mathrm{p}(\varphi)$ je tvaru $\nu y.\varphi_1$, kde $y$ je různé od $x$. Protože platí $f_{\mathrm{p}(\varphi)}(X) = f_{\varphi_1}(X)$, je $f_{\mathrm{p}(\varphi)}$ monotónní zobrazení proměnné $X$ (pro libovolnou hodnotu proměnné $y$).

   (g) Formule $\mathrm{p}(\varphi)$ je tvaru $\mu y.\varphi_1$, kde $y$ je různé od $x$ — viz cvičení 8.8.6.

■

Lemma nám umožňuje podat přesnou definici formulí modálního $\mu$-kalkulu.

**8.5.10 Syntaxe modálního $\mu$-kalkulu** *Formule modálního $\mu$-kalkulu nad množinou atomických formulí At, množinou proměnných Var a množinou akcí* $\Delta$ jsou definovány následujícími pravidly. U každé formule $\varphi$ modálního $\mu$-kalkulu je definována i *množina* $\mathsf{BV}(\varphi)$ *vázaných proměnných ve formuli* $\varphi$ a *množina* $\mathsf{FV}(\varphi)$ *volných proměnných ve formuli* $\varphi$.

1. Každá atomická formule $a$ je formulí modálního $\mu$-kalkulu. Tato formule neobsahuje žádnou proměnnou, volnou ani vázanou: $\mathsf{FV}(a) = \mathsf{BV}(a) = \emptyset$.

2. Každá proměnná $x$ je formulí modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(x) = \emptyset$ a $\mathsf{FV}(x) = \{x\}$.

3. Formule $\mathtt{tt}$ je formulí modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(\mathtt{tt}) = \mathsf{FV}(\mathtt{tt}) = \emptyset$.

4. Jsou-li $\varphi_1$ a $\varphi_2$ formule modálního $\mu$-kalkulu, potom $(\varphi_1 \wedge \varphi_2)$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}((\varphi_1 \wedge \varphi_2)) = \mathsf{BV}(\varphi_1) \cup \mathsf{BV}(\varphi_2)$ a $\mathsf{FV}((\varphi_1 \wedge \varphi_2)) = \mathsf{FV}(\varphi_1) \cup \mathsf{FV}(\varphi_2)$.

5. Jestliže $\varphi$ je formule modálního $\mu$-kalkulu, potom $(\neg\varphi)$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}((\neg\varphi)) = \mathsf{BV}(\varphi)$ a $\mathsf{FV}((\neg\varphi)) = \mathsf{FV}(\varphi)$.

6. Jestliže $\varphi$ je formule modálního $\mu$-kalkulu a $R$ je jméno akce, potom $([R]\varphi)$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}(([R]\varphi)) = \mathsf{BV}(\varphi)$ a $\mathsf{FV}(([R]\varphi)) = \mathsf{FV}(\varphi)$.

7. Je-li $\varphi$ formule modálního $\mu$-kalkulu a je-li $x$ proměnná, která má positivní výskyt ve $\varphi$, potom $(\nu x.\varphi)$ je formule modálního $\mu$-kalkulu. Definujeme $\mathsf{BV}((\nu x.\varphi)) = \mathsf{BV}(\varphi) \cup \{x\}$ a $\mathsf{FV}((\nu x.\varphi)) = \mathsf{FV}(\varphi) \setminus \{x\}$. Prefix $\nu x.$ se tedy chová podobně jako kvantifikátory v predikátové logice — prefix $\nu x.$ *váže* výskyty proměnné $x$ ve formuli $\varphi$.

Budeme opět dodržovat konvenci o vynechávání vnějších závorek. Spojky $\vee$, $\Rightarrow$, $\Leftrightarrow$, modality $\langle R\rangle$ a formuli $\mathtt{ff}$ zavádíme obvyklým způsobem jako syntaktické zkratky. Formuli $\mu x.\varphi$ zavedeme jako syntaktickou zkratku za formuli $\neg\nu x.(\neg\varphi[x \mapsto \neg x])$, kde jako $\varphi[x \mapsto \neg x]$ označujeme formuli vzniklou z formule $\varphi$ náhradou každého *volného* výskytu proměnné $x$ formulí $\neg x$ — viz tvrzení 8.4.2.

*Nová konvence je o vázaných proměnných.* Budeme vždy přepokládat, že formule, které se liší pouze jmény vázaných proměnných jsou totožné (této konvenci se někdy říká $\alpha$*-konverse*). Podle této konvence jsou tedy formule $\nu x.x$ a $\nu y.y$ totožné.

Zavedení $\alpha$-konverse také umožňuje, aby každý výskyt každé proměnné ve formuli $\varphi$ modálního $\mu$-kalkulu byl buď volný nebo vázaný — viz cvičení 8.8.7. Formuli, kde každý výskyt každé proměnné je buď volný nebo vázaný, říkáme *formule s čistým výskytem proměnných.* V dalším budeme předpokládat, že všechny formule mají čistý výskyt proměnných.

**8.5.11 Sémantika modálního $\mu$-kalkulu** Formule modálního $\mu$-kalkulu budeme interpretovat v akčních strukturách typu $\Delta$. *Modální $\mu$-model* $\mathscr{M}$ je trojice $\langle \mathbb{V}, \mathscr{V}, \rho \rangle$, kde:

1. $\mathbb{V}$ je akční struktura typu $\Delta$.
2. $\mathscr{V} : At \to exp(V)$ je valuace atomických formulí.
3. $\rho : Var \to exp(V)$ je kontext proměnných.

Každé formuli $\varphi$ modálního $\mu$-kalkulu je v modálním $\mu$-modelu přiřazena množina vrcholů $\| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle}$ následovně:

1. $\| a \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \mathscr{V}(a)$ pro každé $a \in At$.
2. $\| x \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \rho(x)$ pro každé $x \in Var$.
3. $\| \mathtt{tt} \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = V$.
4. $\| \varphi_1 \wedge \varphi_2 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \| \varphi_1 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} \cap \| \varphi_2 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle}$.
5. $\| \neg \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = V \setminus \| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle}$.
6. $\| [R]\varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \| [R] \|_{\mathbb{V}}(\| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle})$, kde funkce $\| [R] \|_{\mathbb{V}} : exp(V) \to exp(V)$ je definována v $X \subseteq V$ rovností
$$\| [R] \|_{\mathbb{V}}(X) = \{ v \in V \mid \text{je-li } v \xrightarrow{R} v', \text{ pak } v' \in X \}.$$
7. $\| \nu x.\varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \nu f_\varphi$, kde $f_\varphi : exp(V) \to exp(V)$ je funkce definovaná následovně:

   V definici 8.5.10 jsme požadovali positivní výskyt proměnné $x$ ve $\varphi$. Speciálně je tedy proměnná $x$ ve formuli $\varphi$ volná. Potom extense $\varphi$ v $\langle \mathbb{V}, \mathscr{V}, \rho \rangle$ závisí na kontextu proměnných $\rho$. Toho využijeme.

   Funkce $f_\varphi$ přiřadí množině vrcholů $X$ množinu $f_\varphi(X)$ definovanou takto: nejprve změníme kontext proměnných $\rho$ na kontext $\rho'$ tak, aby v kontextu $\rho'$ měla proměnná hodnotu $X$. Ostatní proměnné mají v kontextu $\rho'$ nezměněnou hodnotu.

   Hodnotu funkce $f_\varphi$ v $X$ definujeme jako množinu $\| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho' \rangle}$.

   Definice extense formule $\nu x.\varphi$ je korektní, protože zobrazení $f_\varphi$ je monotónní. V definici 8.5.10 jsme požadovali positivní výskyt proměnné $x$ ve $\varphi$ a podle lemmatu 8.5.9 je zobrazení $f_\varphi$ monotónní.

Extense některých odvozených formulí modálního $\mu$-kalkulu zmíníme v následujícím lemmatu. Jeho důkaz a doplnění vztahů pro extense zbylých odvozených spojek přenecháváme jako cvičení 8.8.8.

**8.5.12 Lemma** *Platí:*

1. $\| \varphi_1 \vee \varphi_2 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \| \varphi_1 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} \cup \| \varphi_1 \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle}$.
2. $\| \langle R \rangle \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \| R \|_{\mathbb{V}}(\| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle})$.
3. $\| \mu x.\varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle} = \mu f_\varphi$, *kde $f_\varphi$ je definováno stejně jako v bodě 7 definice 8.5.10.*

Podobně jako v predikátové logice, budeme v modálním $\mu$-kalkulu mluvit o pravdivosti pouze u *sentencí*, tj. u formulí, které neobsahují žádné volné proměnné. Analogicky jako v modální logice z kapitoly 4 zavedeme pro sentence nejprve lokální pojem vynucení.

**8.5.13 Definice** Ať $\varphi$ je sentence modálního $\mu$-kalkulu. Předpokládejme, že je dána akční struktura $\mathbb{V}$ a valuace atomických proměnných $\mathscr{V}$. Dvojici $\langle \mathbb{V}, \mathscr{V} \rangle$ označíme jako $\mathscr{M}$.

Řekneme, že *sentence $\varphi$ je $\mathscr{M}$-vynucena ve vrcholu $v$* a tento fakt označíme jako $v \Vdash_{\mathscr{M}} \varphi$, pokud $v \in \| \varphi \|_{\langle \mathbb{V}, \mathscr{V}, \rho \rangle}$ pro libovolný kontext $\rho$.

Fakt, že sentence $\varphi$ není $\mathscr{M}$-vynucena ve vrcholu $v$ budeme značit $v \not\Vdash_{\mathscr{M}} \varphi$.

Pro spojky a formule modálního $\mu$-kalkulu platí:

**8.5.14 Lemma** *$\mathscr{M}$-vynucení lze popsat následovně:*

1. $v \Vdash_{\mathscr{M}} \mathtt{tt}$ *platí vždy.*
2. $v \Vdash_{\mathscr{M}} \mathtt{ff}$ *neplatí nikdy.*
3. $v \Vdash_{\mathscr{M}} a$, *pokud platí* $v \in \mathscr{V}(a)$ *pro atomickou formuli* $a$.
4. $v \Vdash_{\mathscr{M}} \varphi_1 \wedge \varphi_2$, *pokud platí* $v \Vdash_{\mathscr{M}} \varphi_1$ *a současně* $v \Vdash_{\mathscr{M}} \varphi_2$.
5. $v \Vdash_{\mathscr{M}} \varphi_1 \vee \varphi_2$ *platí právě tehdy, když platí* $v \Vdash_{\mathscr{M}} \varphi_1$ *nebo* $v \Vdash_{\mathscr{M}} \varphi_2$.
6. $v \Vdash_{\mathscr{M}} \neg\varphi_1$, *pokud neplatí* $v \Vdash_{\mathscr{M}} \varphi_1$.
7. $v \Vdash_{\mathscr{M}} \varphi_1 \Rightarrow \varphi_2$ *platí právě tehdy, když platí: je-li* $v \Vdash_{\mathscr{M}} \varphi_1$, *pak* $v \Vdash_{\mathscr{M}} \varphi_2$.
8. $v \Vdash_{\mathscr{M}} [R]\varphi_1$, *pokud pro libovolnou akci* $v \xrightarrow{R} v'$ *platí* $v' \Vdash_{\mathscr{M}} \varphi_1$.
9. $v \Vdash_{\mathscr{M}} \langle R\rangle\varphi$ *platí právě tehdy, když existuje vrchol* $v'$ *a akce* $v \xrightarrow{R} v'$ *tak, že* $v' \Vdash_{\mathscr{M}} \varphi$.
10. $v \Vdash_{\mathscr{M}} \nu x.\varphi_1$, *pokud platí* $v \in \nu f_{\varphi_1}$, *kde* $f_{\varphi_1}$ *je definováno stejně jako v bodě 7 definice 8.5.10.*
11. $v \Vdash_{\mathscr{M}} \mu x.\varphi_1$, *pokud platí* $v \in \mu f_{\varphi_1}$, *kde* $f_{\varphi_1}$ *je definováno stejně jako v bodě 7 definice 8.5.10.*

Důkaz. Viz cvičení 8.8.9. ■

Podobně jako v Henessyho-Milnerově modální logice zavedeme globální pojem pravdivosti.

**8.5.15 Definice** Řekneme, že *sentence modálního $\mu$-kalkulu $\varphi$ je pravdivá v $\mathscr{M} = \langle \mathbb{V}, \mathscr{V}\rangle$* a tento fakt značíme $\models_{\mathscr{M}} \varphi$, jestliže platí $v \Vdash_{\mathscr{M}} \varphi$ pro každý vrchol $v$. Fakt, že formule $\varphi$ *není pravdivá v modelu $\mathscr{M}$*, značíme $\not\models_{\mathscr{M}} \varphi$.

Samozřejmě, i v modálním $\mu$-kalkulu lze zavést pojem *tautologie* pro sentenci $\varphi$ pravdivou v každém $\mathscr{M}$. Vzniká přirozená otázka, zda lze pro tautologie modálního $\mu$-kalkulu najít analogii deduktivního systému **K** z kapitoly 5 a dokázat větu o úplnosti. Existence takového deduktivního systému je však dosud *otevřeným problémem*.

## 8.6 Vlastnosti Liveness a Safety

V tomto odstavci ukážeme, že formulemi modálního $\mu$-kalkulu lze popsat dvě důležité vlastnosti procesů Liveness a Safety:

- Vlastnost Liveness intuitivně popisuje situaci, kdy systém jednou musí přejít do „dobrého“ stavu.
- Vlastnost Safety intuitivně popisuje situaci, kdy systém nikdy nepřejde do „špatného“ stavu.

Než se budeme těmto dvěma vlastnostem věnovat v plné obecnosti, uvedeme několik příkladů.

**8.6.1 Příklad** Intuitivní význam sentencí $\mu x.[R]x$ a $\nu x.[R]x$.
Označme formuli $[R]x$ jako $\alpha$. Pro popis vynucení sentencí $\mu x.[R]x$ a $\nu x.[R]x$ bude důležité znát příslušnou monotónní funkci $f_\alpha$.

Funkce $f_\alpha$ přiřadí množině vrcholů $X$ množinu $\| [R] \|_{\mathbb{V}}(X)$ což je množina

$$\{v \in V \mid \text{je-li } v \xrightarrow{R} v', \text{ potom } v' \in X\}.$$

Jedná se tedy o funkci $r$ z příkladu 4 v 8.3.7. Tam jsme ukázali, že:

1. Nejmenším pevným bodem zobrazení $r$ je nejmenší množina vrcholů $M$ s vlastností

([$R$]-B) Je-li vrchol $v$ takový, že pro libovolnou akci $v \xrightarrow{R} v'$ je $v' \in M$, potom $v \in M$.

Sentence $\mu x.[R]x$ je tedy vynucena přesně v těch vrcholech, patřících do všech množin, které uzavřeny na akci $R$ „směrem dozadu“. Jde o vlastnost Safety v následujícím smyslu: množina $\mu r$ je nejmenší množina vrcholů, ze které se žádným prováděním akce $R$ nelze dostat ven.

2. Největším pevným bodem zobrazení $r$ je největší množina vrcholů $M$ s vlastností

([$R$]-F) Je-li vrchol $v$ v množině $M$, pak pro libovolnou akci $v \xrightarrow{R} v'$ je $v' \in M$.

$\nu f_\alpha$ je tedy největší množina vrcholů, která je uzavřena na akci $R$ „směrem dopředu“.

V příkladu 4 z 8.3.7 jsme ukázali, že největší taková množina je množina všech vrcholů, tedy $\mu f_\alpha = V$. Podle definice tedy $v \Vdash_{\mathscr{M}} \nu x.[R]x$ platí právě tehdy, když $v \in V$.

Uzavíráme tedy, že $v \Vdash_{\mathscr{M}} \nu x.[R]x$ platí vždy.

*Sentence $\nu x.[R]x$ má stejný význam jako formule* `tt`.

Shrnuto: sentence $\nu x.[R]x$ není zajímavá — má stejný význam jako formule `tt`. Sentence $\mu x.[R]x$ však zajímavá je — vyjadřuje Safety vlastnost pro provádění akce $R$.

**8.6.2 Příklad** Intuitivní význam sentencí $\mu x.\langle R\rangle x$ a $\nu x.\langle R\rangle x$.
Opět označme formuli $\langle R\rangle x$ jako $\alpha$. Pro popis vynucení sentencí $\mu x.\langle R\rangle x$ a $\nu x.\langle R\rangle x$ bude důležité znát příslušnou monotónní funkci $f_\alpha$.

Funkce $f_\alpha$ přiřadí množině vrcholů $X$ množinu $\| \langle R\rangle \|_{\mathbb{V}}(X)$ což je množina

$$\{v \in V \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in X\}.$$

1. Nejmenším pevným bodem zobrazení $f_\alpha$ je nejmenší množina vrcholů $M$ s vlastností

($\langle R\rangle$-B) Je-li vrchol $v$ takový, že existuje akce $v \xrightarrow{R} v'$ tak, že $v' \in M$, potom $v \in M$.

Nejmenší taková množina je zřejmě prázdná množina.

Podle definice tedy $v \Vdash_{\mathscr{M}} \mu x.\langle R\rangle x$ platí právě tehdy, když $v \in \emptyset$.

Uzavíráme, že $v \Vdash_{\mathscr{M}} \mu x.\langle R\rangle x$ neplatí nikdy. *Sentence $\mu x.\langle R\rangle x$ má stejný význam jako formule* `ff`.

2. Největším pevným bodem zobrazení $r$ je největší množina vrcholů $M$ s vlastností

($\langle R\rangle$-F) Je-li vrchol $v$ v množině $M$, pak existuje akce $v \xrightarrow{R} v'$ tak, že je $v' \in M$.

$\nu f_\alpha$ je tedy největší množina vrcholů, která je umožňuje vykonat akci $R$ a zůstat v této množině.

Podle definice $v \Vdash_{\mathscr{M}} \nu x.\langle R\rangle x$ platí právě tehdy, když $v \in \nu f_\alpha$. Sentence $\nu x.\langle R\rangle x$ tedy vyjadřuje vlastnost Liveness v následujícím smyslu: množina $\mu r$ je největší množina těch vrcholů, které umožňují provedení akce $R$ a zůstat v této množině.

Shrnuto: sentence $\mu x.\langle R\rangle x$ není zajímavá — má stejný význam jako formule `ff`. Sentence $\nu x.\langle R\rangle x$ však zajímavá je — vyjadřuje Liveness vlastnost pro provádění akce $R$.

Předchozí dva příklady naznačují, jak definovat sentence pro vlastnosti Safety a Liveness obecně.

1. Vlastnost Safety: zvolme sentenci $\varphi$ modálního $\mu$-kalkulu, která neobsahuje proměnnou $x$. Intuitivně má sentence $\varphi$ význam takový, že při provádění akce $R$ chceme, aby stále platila.

Sentence $\mu x.(\varphi \wedge [R]x)$ je pak vynucena ve vrcholech takových, že při jakémkoli vykonání akce $R$ skončíme ve vrcholu, kde je $\varphi$ pravdivá. Jinak: při jakémkoli „běhu“ akcí $R$ stále platí $\varphi$.

Situaci lze popsat i sloganem *Nikdy nebude špatně*. Špatně je, když formule $\varphi$ neplatí.

2. Vlastnost Liveness: zvolme sentenci $\varphi$ modálního $\mu$-kalkulu, která neobsahuje proměnnou $x$. Intuitivně má sentence $\varphi$ význam takový, že při chceme, aby po nějakém provádění akce $R$ platila.

Sentence $\nu x.(\varphi \vee \langle R\rangle x)$ je pak vynucena ve vrcholech takových, že existuje „běh“ akcí $R$ takový, že se buď dostaneme do vrcholu, kde je $\varphi$ pravdivá nebo s touto nadějí můžeme akcí $R$ dále pokračovat.

Situaci lze popsat i sloganem *Jednou bude dobře*. Dobře je, když formule $\varphi$ platí.

K vlastnostem Liveness a Safety se ještě vrátíme v následujícím odstavci.

**8.6.3 Poznámka** Poznamenejme, že někdy se za Liveness vlastnost bere sentence $\nu x.(\varphi \vee (\langle R \rangle \mathtt{tt} \wedge [R]x))$. Tuto sentenci lze intuitivně popsat sloganem *Vždy lze udělat něco pro to, aby bylo dobře.* Viz cvičení 8.8.13.

## 8.7 Konečné aproximace modálních $\mu$-formulí

Pochopit význam obecné modální $\mu$-formule není snadné. V případě, že pracujeme na konečné akční struktuře, lze význam $\mu$-formule pochopit pomocí jejích konečných aproximací. Začneme nejprve vyšetřováním konečných aproximací pevných bodů.

Podle Tarského věty pro monotónní zobrazení $f : \mathbb{X} \to \mathbb{X}$ na úplném svazu platí:

1. Zobrazení $f$ má nejmenší pevný bod počítaný jako infimum prefixních bodů zobrazení $f$, tj. $\mu f = \inf \mathsf{Pre}(f)$.
2. Zobrazení $f$ má největší pevný bod počítaný jako supremum postfixních bodů zobrazení $f$, tj. $\nu f = \sup \mathsf{Post}(f)$.

V tomto odstavci ukážeme, že nejmenší a největší pevný bod monotónního zobrazení na úplném svazu lze nalézt i jinak. Musíme se však omezit na situace, kdy úplný svaz $\mathbb{X}$ je konečná množina. Pak dokonce stačí předpokládat, že $\mathbb{X}$ je pouze svaz — jeho úplnost plyne z jeho konečnosti, viz cvičení 8.8.11.

Základem našich úvah je platnost následujícího jednoduchého tvrzení:

**8.7.1 Lemma** *Ať $\mathbb{X}$ je libovolný (ne nutně konečný) poset. Předpokládejme, že $f : \mathbb{X} \to \mathbb{X}$ je monotónní zobrazení. Potom platí:*

1. *Je-li $x$ prefixní bod $f$, potom i $f(x)$ je prefixní bod $f$. Navíc platí nerovnost $f(x) \sqsubseteq x$.*
2. *Je-li $x$ postfixní bod $f$, potom i $f(x)$ je postfixní bod $f$. Navíc platí nerovnost $x \sqsubseteq f(x)$.*

Důkaz. Dokážeme pouze první trzení, druhé lze dokázat analogicky. Ať $x$ je prefixní bod. Platí tedy nerovnost $f(x) \sqsubseteq x$. Protože $f$ je monotónní zobrazení, platí nerovnost $f(f(x)) \sqsubseteq f(x)$. To ale znamená, že $f(x)$ je prefixní bod zobrazení $f$. ■

Z předchozího tvrzení plyne, že začneme-li s prefixním bodem $x$ pro zobrazení $f$, lze vytvořit klesající posloupnost prefixních bodů $x_n$ definovanou následovně:

$$x_0 = x \qquad x_{n+1} = f(x_n)$$

Analogicky, stejně definovaná posloupnost $x_n$ je rostoucí posloupností postfixních bodů zobrazení $f$, pokud jsme začali s postfixním bodem $x$.

Uvědomme si, že pro $f$ existuje vždy největší prefixní bod a nejmenší postfixní bod:

1. Největším prefixním bodem je největší prvek $\top$ ve svazu $\mathbb{X}$: platí totiž nerovnost $f(\top) \sqsubseteq \top$.
2. Nejmenším postfixním bodem je nejmenší prvek $\bot$ ve svazu $\mathbb{X}$: platí totiž nerovnost $\bot \sqsubseteq f(\bot)$.

Máme tedy vždy k disposici kanonickou volbu počáteční hodnoty posloupnosti $x_n$. Navíc, protože předpokládáme, že svaz $\mathbb{X}$ je konečný, platí následující věta:

**8.7.2 Věta (Konečné aproximace pevných bodů)** *Ať $f : \mathbb{X} \to \mathbb{X}$ je monotónní zobrazení na konečném svazu. Definujme posloupnosti $x_n$ a $y_n$ takto:*

$$x_0 = \bot \qquad x_{n+1} = f(x_n) \qquad y_0 = \top \qquad y_{n+1} = f(y_n)$$

*Potom platí:*

1. *Platí $\mu f = \sup \{x_n \mid n \geq 0\}$.*
2. *Platí $\nu f = \inf \{y_n \mid n \geq 0\}$.*

Důkaz. Dokážeme první tvrzení, druhé lze dokázat analogicky. Označme $s = \sup\{x_n \mid n \geq 0\}$. Protože je svaz $\mathbb{X}$ konečný, nemůže posloupnost $x_n$ být ostře rostoucí pro všechna $n$. Existuje tedy nejmenší $n_0$ takové, že $x_n = x_{n_0}$ pro všechna $n \geq n_0$. To znamená, že $s = x_{n_0}$.

Protože platí, že $x_{n_0+1} = x_{n_0} = s$, je podle definice posloupnosti $x_n$ prvek $s$ pevný bod funkce $f$. Potom musí platit nerovnost $\mu f \sqsubseteq s$, protože $\mu f$ je nejmenší pevný bod.

Abychom ukázali, že platí nerovnost $s \sqsubseteq \mu f$, stačí ukázat, že platí nerovnost $x_n \sqsubseteq \mu f$ pro všechna $n \geq 0$.

- Nerovnost $x_0 \sqsubseteq \mu f$ platí triviálně, protože $x_0 = \bot$.
- Předpokládáme-li, že platí $x_n \sqsubseteq \mu f$, potom platí $f(x_n) \sqsubseteq f(\mu f)$. Protože $f(x_n) = x_{n+1}$, ukázali jsme, že $x_{n+1} \sqsubseteq \mu f$. (Využili jsme toho, že platí rovnost $f(\mu f) = \mu f$.)

■

Předchozí větu lze nyní snadno přenést na pravdivost sentencí modálního $\mu$-kalkulu v konečných modelech. Nejprve budeme definovat konečné aproximace sentencí $\mu x.\varphi$ a $\nu x.\varphi$.

**8.7.3 Definice** *Posloupnosti konečných aproximací* sentencí $\mu x.\varphi$ a $\nu x.\varphi$ jsou definovány takto:

1. Pro sentenci $\mu x.\varphi$:
$$\mu^0 x.\varphi = \mathtt{ff} \qquad \mu^{n+1} x.\varphi = \varphi[x \mapsto \mu^n x.\varphi]$$
kde $\varphi[x \mapsto \mu^n x.\varphi]$ značí formuli, kterou jsme obdrželi z formule $\varphi$ náhradou každého volného výskytu proměnné $x$ formulí $\mu^n x.\varphi$.
2. Pro sentenci $\nu x.\varphi$:
$$\nu^0 x.\varphi = \mathtt{tt} \qquad \nu^{n+1} x.\varphi = \varphi[x \mapsto \nu^n x.\varphi]$$
kde $\varphi[x \mapsto \nu^n x.\varphi]$ značí formuli, kterou jsme obdrželi z formule $\varphi$ náhradou každého volného výskytu proměnné $x$ formulí $\nu^n x.\varphi$.

**8.7.4 Věta** *Ať $\mathbb{V}$ je konečná akční struktura a $\mathscr{V}$ je valuace atomických proměnných. Označme $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$. Potom platí:*

1. *$v \Vdash_{\mathscr{M}} \mu x.\varphi$ právě tehdy, když existuje $n \geq 0$ takové, že platí $v \Vdash_{\mathscr{M}} \mu^n x.\varphi$.*
2. *$v \Vdash_{\mathscr{M}} \nu x.\varphi$ právě tehdy, když pro všechna $n \geq 0$ platí $v \Vdash_{\mathscr{M}} \nu^n x.\varphi$.*

Důkaz. Dokážeme pouze první tvrzení, druhé lze dokázat analogicky.

Protože předpokládáme, že akční struktura $\mathbb{V}$ je konečná, je konečná i množina $exp(V)$. Funkce $f_\varphi$ definovaná stejně jako v bodě 7 definice 8.5.10 je tedy monotónní funkce na konečném svazu. Její nejmenší pevný bod $\mu f_\varphi$ lze tedy hledat pomocí věty 8.7.2 jako $\bigcup\{X_n \mid n \geq 0\}$, kde $X_n$ je posloupnost množin vrcholů vytvořená následovně:
$$X_0 = \emptyset \qquad X_{n+1} = f_\varphi(X_n).$$
První tvrzení plyne nyní z toho, že $v \Vdash_{\mathscr{M}} \mu^n x.\varphi$ platí právě tehdy, když $v \in X_n$. ■

Větu 8.7.4 nyní použijeme na formule popisující vlastnosti Safety a Liveness z odstavce 8.6.

**8.7.5 Příklad** Předpokládejme, že $\mathbb{V}$ je konečná akční struktura a $\mathscr{V}$ je valuace atomických formulí. Označme $\mathscr{M} = \langle \mathbb{V}, \mathscr{V} \rangle$.

1. Pro vlastnost Safety volíme sentenci $\varphi$ modálního $\mu$-kalkulu, která neobsahuje proměnnou $x$. Vlastnost Safety pak popisujeme sentencí $\mu x.(\varphi \wedge [R]x)$. Popíšeme konečné aproximace této sentence:
   (a) Nultá aproximace je sentence $\mathtt{ff}$. Tato aproximace není vynucena nikde.
   (b) První aproximace je sentence $(\varphi \wedge [R]x)[x \mapsto \mathtt{ff}] = \varphi \wedge [R]\mathtt{ff}$. Tato aproximace je vynucena přesně v těch vrcholech, kde platí $\varphi$ a nelze vykonat akci $R$.

(c) Druhá aproximace je sentence $(\varphi \wedge [R]x)[x \mapsto \varphi \wedge [R]\mathtt{ff}] = \varphi \wedge [R](\varphi \wedge [R]\mathtt{ff})$. Tato aproximace je vynucena přesně v těch vrcholech, kde platí $\varphi$ a každé vykonání akce $R$ skončí ve vrcholu, kde platí $\varphi$ a nelze vykonat akci $R$.

Takto lze pokračovat dále. Stejné aproximace sentence $\mu x.(\varphi \wedge [R]x)$ jde samozřejmě vytvořit v libovolné (ne nutně konečné) akční struktuře. Je to však právě konečnost $\mathbb{V}$, která zaručuje platnost vztahu $v \Vdash_{\mathscr{M}} \mu x.(\varphi \wedge [R]x)$ právě tehdy, když existuje $n \geq 0$ takové, že platí $v \Vdash_{\mathscr{M}} \mu^n x.(\varphi \wedge [R]x)$.

2. Pro vlastnost Liveness volíme sentenci $\varphi$ modálního $\mu$-kalkulu, která neobsahuje proměnnou $x$. Vlastnost Liveness pak popisujeme sentencí $\nu x.(\varphi \vee \langle R\rangle x)$. Popíšeme konečné aproximace této sentence:

   (a) Nultá aproximace je sentence $\mathtt{tt}$. Tato aproximace je vynucena všude.

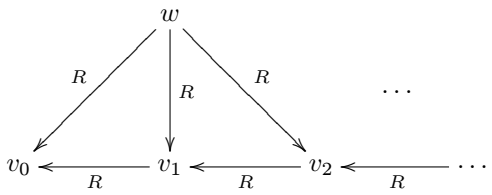
   (b) První aproximace je sentence $(\varphi \vee \langle R\rangle x)[x \mapsto \mathtt{tt}] = \varphi \vee \langle R\rangle\mathtt{tt}$. Tato aproximace je vynucena přesně v těch vrcholech, kde platí $\varphi$ nebo lze vykonat akci $R$.

   (c) Druhá aproximace je sentence $(\varphi \vee \langle R\rangle x)[x \mapsto \varphi \vee \langle R\rangle\mathtt{tt}] = \varphi \vee \langle R\rangle(\varphi \vee \langle R\rangle\mathtt{tt})$. Tato aproximace je vynucena přesně v těch vrcholech, kde platí $\varphi$ nebo lze vykonat akci $R$ a skončit ve vrcholu, kde platí $\varphi$ nebo lze vykonat akci $R$.

   Takto lze pokračovat dále. Stejné aproximace sentence $\nu x.(\varphi \vee \langle R\rangle x)$ jde samozřejmě vytvořit v libovolné (ne nutně konečné) akční struktuře. Je to však právě konečnost $\mathbb{V}$, která zaručuje platnost vztahu $v \Vdash_{\mathscr{M}} \nu x.(\varphi \vee \langle R\rangle x)$ právě tehdy, když pro všechna $n \geq 0$ platí $v \Vdash_{\mathscr{M}} \nu^n x.(\varphi \vee \langle R\rangle x)$.

Jak ukazuje následující příklad, je konečnost akční struktury pro korektní použití konečných aproximací nutná.

**8.7.6 Příklad** $\mathbb{V}$ je následující akční struktura:




V jazyce CCS je tedy $\mathbb{V}$ popsána jako

$$\begin{aligned} w &\stackrel{\text{def}}{=} \sum_{n=0}^{\infty} R.v_n \\ v_{n+1} &\stackrel{\text{def}}{=} R.v_n \quad \text{pro každé } n \end{aligned}$$

Předpokládejme, že $\mathscr{V}$ je nějaká valuace atomických formulí a označme dvojici $\langle \mathbb{V}, \mathscr{V}\rangle$ jako $\mathscr{M}$. Ukážeme následující dvě tvrzení (porovnejte s druhým tvrzením věty 8.7.4):

1. Pro každé $n \geq 0$ platí $w \Vdash_{\mathscr{M}} \nu^n x.\langle R\rangle x$.

2. Neplatí $w \Vdash_{\mathscr{M}} \nu x.\langle R\rangle x$.

K důkazu prvního tvrzení stačí ukázat, že $w \in \| \nu^n x.\langle R\rangle x \|_{\mathscr{M}}$ pro každé $n$. Označme jako $X_n$ množinu $\| \nu^n x.\langle R\rangle x \|_{\mathscr{M}}$. Dále označme jako $f$ monotónní zobrazení odpovídající formuli $\langle R\rangle x$. Jde zřejmě o zobrazení definované jako $f(X) = \| \langle R\rangle \|_{\mathscr{M}}(X)$. Víme, že platí $X_0 = V$ a $X_{n+1} = f(X_n)$ pro všechna $n \geq 0$.

Ukážeme, že pro každé $n \geq 1$ platí $X_n = V \setminus \{v_0, \ldots, v_{n-1}\}$. Budeme postupovat indukcí podle $n$.

- Je-li $n = 1$, je $X_1 = f(V) = \{v \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in V\}$. Poslední množina je přesně $V \setminus \{v_0\}$.

- Předpokládejme, že pro $n \geq 1$ platí $X_n = V \setminus \{v_0, \ldots, v_{n-1}\}$.

  Potom $X_{n+1} = f(X_n) = \{v \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in X_n\}$. Poslední množina je přesně množina $V \setminus \{v_0, \ldots, v_n\}$.

Ukázali jsme tedy, že $w \in X_n$ pro všechna $n \geq 0$.

Abychom ukázali, že neplatí $w \Vdash_{\mathscr{M}} \nu x.\langle R\rangle x$, stačí spočítat největší pevný bod $\nu f$ funkce $f$ a ukázat, že $w \notin \nu f$. To ukážeme tak, že předvedeme, že jediným pevným bodem zobrazení $f$ je prázdná množina.

- Prázdná množina je pevným bodem zobrazení $f$, protože platí

$$f(\emptyset) = \{v \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in \emptyset\} = \emptyset.$$

- Předpokládejme, že existuje *neprázdná* množina $X$ taková, že $f(X) = X$. Odvodíme spor.

  Nejprve si uvědomme, že v množině $X$ musí být nutně nějaký vrchol $v_n$. Kdyby žádný z vrcholů $v_n$ v množině $X$ nebyl, musel by v $X$ ležet pouze vrchol $w$. Protože platí $X = f(X)$, existuje akce $R$ vedoucí z vrcholu $w$ do nějakého vrcholu množiny $X$. To ale v případě $X = \{w\}$ není možné.

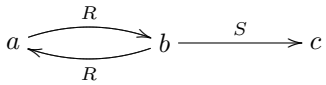
  Víme tedy, že musí existovat $n$ tak, že $v_n \in X$. Vezměme nejmenší takové $n$ a povšimněme si, že nutně musí platit $n > 0$. Vrchol $v_0$ totiž nemůže vykonat žádnou akci, a proto v množině $X$ být nemůže.

  Protože $v_n$ je v množině $X$ a protože $X = f(X)$, musí být v množině $X$ i vrchol $v_{n-1}$. To je spor s naší volbou čísla $n$.

Jediným (a tedy i největším) pevným bodem zobrazení $f$ je prázdná množina, a proto sentence $\nu x.\langle R\rangle x$ není ve vrcholu $w$ vynucena.

Předvedeme nyní netriviální použití konečných aproximací a věty 8.7.4.

**8.7.7 Příklad** $\mathbb{V}$ je následující akční struktura:




Ať $\mathscr{V}$ je nějaká valuace a označme dvojici $\langle \mathbb{V}, \mathscr{V}\rangle$ jako $\mathscr{M}$. Ukážeme, že platí $a \Vdash_{\mathscr{M}} \nu x.\langle R\rangle x$. Protože $\mathbb{V}$ je konečná akční struktura, můžeme ke spočítání největšího pevného bodu funkce

$$f : X \mapsto \| \langle R\rangle \|_{\mathscr{M}}(X)$$

použít metodu konečných aproximací. Definujeme tedy $X_0 = \{a, b, c\}$ a $X_{n+1} = f(X_n)$ pro všechna $n \geq 0$.

První tři členy posloupnosti $X_n$ vypadají následovně:

- $X_0 = \{a, b, c\}$.
- $X_1 = f(X_0) = \{v \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in X_0\} = \{a, b\}$.
- $X_2 = f(X_1) = \{v \mid \text{existuje } v \xrightarrow{R} v' \text{ tak, že } v' \in X_1\} = \{a, b\}$.

Dále již nemá cenu pokračovat, protože se posloupnost $X_n$ stabilisovala. Největším pevným bodem zobrazení $f$ je množina $\{a, b\}$. Protože $a \in \{a, b\}$, platí $a \Vdash_{\mathscr{M}} \nu x.\langle R\rangle x$.

Aproximace sentence $\nu x.\langle R\rangle x$ nyní využijeme k vyjádření toho, co sentence $\nu x.\langle R\rangle x$ opravdu „říká“: uvědomme si, že $n$-tá aproximace sentence $\nu x.\langle R\rangle x$ je přesně formule $\underbrace{\langle R\rangle \ldots \langle R\rangle}_{n-\text{krát}}\mathtt{tt}$. Proto sentence $\nu x.\langle R\rangle x$ vyjadřuje *nekonečné chování*: ve vrcholu $v$ je vynucena sentence $\nu x.\langle R\rangle x$ přesně tehdy, když je pro každé $n \geq 0$ možno z vrcholu $v$ vykonat „běh“ akcí $R$ délky $n$.

Jako ilustraci toho, že pro sentenci modálního $\mu$-kalkulu umíme rozhodnout její pravdivost či nepravdivost v daném vrcholu, aniž bychom dané sentenci „rozuměli“, uvádíme následující příklad:

$$b \Vdash_{\mathscr{M}} \nu x.\mu y.(\langle R\rangle((\langle S\rangle \mathtt{tt} \wedge x) \vee y)).$$

Sestavíme příslušnou monotónní funkci, najdeme její největší pevný bod a ukážeme, že vrchol $b$ je prvkem tohoto největšího pevného bodu.

Označme $M = \| \nu x.\mu y.(\langle R\rangle((\langle S\rangle \mathtt{tt} \wedge x) \vee y)) \|_{\mathscr{M}}$. Potom $M = \nu f$, kde $f$ je zobrazení:

$$f : X \mapsto \mu g_X$$

a pro každou množinu vrcholů $X$ je $g_X$ zobrazení

$$g_X : Y \mapsto \| \langle R\rangle \|_{\mathbb{V}}((\| \langle S\rangle \|_{\mathscr{M}}(V) \cap X) \cup Y).$$

Nejprve předpis pro zobrazení $g_X$ upravíme. Protože $(\| \langle S\rangle \|_{\mathscr{M}}(V) = \{b\}$, dostáváme, že $g_X$ je zobrazení

$$g_X : Y \mapsto \| \langle R\rangle \|_{\mathbb{V}}((\{b\} \cap X) \cup Y).$$

Pro pevné $X$ nyní spočítáme $\mu g_X$. Budeme postupovat metodou konečných aproximací.

- Nultou aproximací $\mu g_X$ je prázdná množina.
- První aproximací $\mu g_X$ je množina $g_X(\emptyset)$, neboli množina

  $$\| \langle R\rangle \|_{\mathbb{V}}((\{b\} \cap X) \cup \emptyset) = \| \langle R\rangle \|_{\mathbb{V}}(\{b\} \cap X).$$

  Rozlišíme dva případy:

  1. Platí $b \in X$. V tomto případě je $g_X(\emptyset) = \| \langle R\rangle \|_{\mathbb{V}}(\{b\}) = \{a\}$.
  2. Platí $b \notin X$. V tomto případě $g_X(\emptyset) = \| \langle R\rangle \|_{\mathbb{V}}(\emptyset) = \emptyset$. Prázdná množina je nejmenším pevným bodem zobrazení $g_X$.
- Druhou aproximaci $\mu g_X$ budeme již počítat pouze pro případ $b \in X$. Tato aproximace je množina $g_X(\{a\})$. Podle definice zobrazení $g_X$ jde o množinu

  $$\| \langle R\rangle \|_{\mathbb{V}}((\{b\} \cap X) \cup \{a\}) = \| \langle R\rangle \|_{\mathbb{V}}(\{a, b\}) = \{a, b\}.$$

  V tomto případě je nejmenším pevným bodem zobrazení $g_X$ množina $\{a, b\}$.

Protože jsme pro každé $X$ spočetli $\mu g_X$, můžeme přepsat předpis pro zobrazení $f$ následovně:

$$f : X \mapsto \begin{cases} \emptyset, & \text{v případě } b \notin X \\ \{a, b\}, & \text{v případě } b \in X. \end{cases}$$

Největším pevným bodem tohoto zobrazení je zřejmě množina $\{a, b\}$. Ukázali jsme, že $b \in \nu f$, neboli platí

$$b \Vdash_{\mathscr{M}} \nu x.\mu y.(\langle R\rangle((\langle S\rangle \mathtt{tt} \wedge x) \vee y))$$

což jsme chtěli ukázat.

## 8.8 Cvičení

**8.8.1 Cvičení** Dokažte druhou část věty 8.3.1.

**8.8.2 Cvičení** Ukažte, že zobrazení analogická k zobrazením z příkladu 8.3.3 jsou monotónní v libovolném svazu $\langle X, \sqcap, \sqcup\rangle$.

1. *Konstatní zobrazení je monotónní.* Zvolte pevně prvek $p \in X$. Zobrazení $f_p : X \to X$, definované předpisem $f_p(x) = p$, je monotónní.
2. *Infimum s pevným prvkem je monotónní.* Zvolte pevně prvek $p \in X$. Zobrazení $g_p : X \to X$, definované předpisem $g_p(x) = p \sqcap x$, je monotónní.
3. *Supremum s pevným prvkem je monotónní.* Zvolte pevně prvek $p \in X$. Zobrazení $h_p : X \to X$, definované předpisem $h_p(x) = p \sqcup x$, je monotónní.

Ukažte, že jestliže svaz $\langle X, \sqcap, \sqcup\rangle$ je komplementární, potom *zobrazení komplementu monotónní není*. Zobrazení komplementu je zobrazení $h : X \to X$ takové, že $h(x) = x^{-1}$.

**8.8.3 Cvičení** Popište extensi odvozených logických spojek, tj. popište zobrazení $\| \vee \|_{\mathbb{V}}$, $\| \Rightarrow \|_{\mathbb{V}}$ a $\| \Leftrightarrow \|_{\mathbb{V}}$.

**8.8.4 Cvičení** Podejte přesnou definici pojmu syntaktického stromu pro formule vytvořené podle pravidel uvedených v 8.5.4.

**8.8.5 Cvičení** Ukažte, že každou formuli $\varphi$ vytvořenou podle pravidel uvedených v 8.5.4 existuje formule $\mathrm{p}(\varphi)$ vytvořená podle pravidel uvedených v 8.5.4 taková, že platí následující dvě podmínky:

1. Pro libovolné $\mathscr{M}$ a $\rho$ platí rovnost $\| \mathrm{p}(\varphi) \|_{\mathscr{M},\rho} = \| \varphi \|_{\mathscr{M},\rho}$.
2. Ve formuli $\mathrm{p}(\varphi)$ se vyskytují pouze spojky disjunkce, konjunkce, boxy, diamanty, prefixy $\nu$ a $\mu$ a spojka negace se vyskytuje pouze těsně před atomickými formulemi a proměnnými.

Formuli $\mathrm{p}(\varphi)$ se říká *positivní forma formule* $\varphi$.

**8.8.6 Cvičení** Dokončete důkaz lemmatu 8.5.9, tj. proberte případy kdy $\mathrm{p}(\varphi)$ je tvaru $\varphi_1 \vee \varphi_2$, $\langle R\rangle\varphi_1$ a $\mu y.\varphi_1$, kde $y$ je různé od $x$.

**8.8.7 Cvičení** Ukažte, že každou formuli $\varphi$ modálního $\mu$-kalkulu lze napsat tak, aby se ve $\varphi$ každá proměnná vyskytovala buď jako volná nebo vázaná.

**8.8.8 Cvičení** Dokažte lemma 8.5.12 a odvoďte extense spojek $\Rightarrow$, $\Leftrightarrow$ a formule `ff`.

**8.8.9 Cvičení** Dokažte lemma 8.5.14.

**8.8.10 Cvičení** Formule $b$ je Safety vlastnost pro provádění těla cyklu `while` $b$ `do` $q$ `od`. Vysvětlete to.

**8.8.11 Cvičení** Ať $\langle X, \sqsubseteq\rangle$ je svaz, kde množina $X$ je konečná. Ukažte, že potom $\langle X, \sqsubseteq\rangle$ je úplný svaz.

**8.8.12 Cvičení** Ukažte, že v akční struktuře $\mathbb{V}$ z příkladu 8.1.1 platí, že sentence $\nu x.\langle R\rangle x$ je vynucena (v libovolném modelu) přesně ve vrcholu $w$.

**8.8.13 Cvičení** Napište konečné aproximace sentence $\nu x.(\varphi \vee (\langle R\rangle\mathtt{tt} \wedge [R]x))$ z poznámky 8.8.13.

**8.8.14 Cvičení** Napište konečné aproximace sentence $\nu x.(([q]x \wedge b) \vee (\varphi \wedge \neg b))$ a označte tyto sentence jako $\varphi_n$. Sentencím $\varphi_n$ dejte intuitivní význam.

Sentence $\nu x.(([q]x \wedge b) \vee (\varphi \wedge \neg b))$ je nejslabší liberální prepodmínkou programu `while` $b$ `do` $q$ `od` vzhledem k formuli $\varphi$.

# Kapitola 9

# Jazyk CCS

V této kapitole rozšíříme notaci kalkulu CCS, kterou jsme zavedli v odstavci 3.2. Jazyk CCS dovoluje elegantní algebraický popis procesů a programů, které spolu mohou nějakým způsobem komunikovat. Doposud jsme používali jazyk CCS pouze jako notaci, v této kapitole definujeme jazyk CCS přesně a vysvětlíme základní vlastnosti pojmu silná bisimilarita. Pro podrobný popis CCS jako jazyka komunikujících systémů a řadu motivačních příkladů odkazujeme na knihu [Mi89].

## 9.1 Syntaxe CCS

Připomeňme, že v odstavci 3.2 jsme zavedli fragment jazyka CCS, který nám sloužil k zadávání akčních struktur.
Syntaktická pravidla tohoto fragmentu jsou:

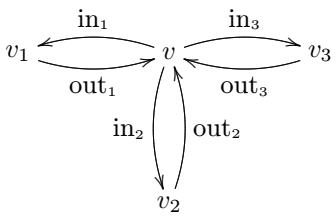
1. *Term jazyka CCS* je buď $v$ nebo $R.v$, kde $v$ je prvek nějaké množiny $V$ a $R$ je prvek množiny $\Delta$.
2. *Fráze jazyka CCS* je zápis
$$v \stackrel{\text{def}}{=} R.w$$
(čteme: $v$ je definováno jako $R$ tečka $w$) nebo
$$v \stackrel{\text{def}}{=} R_1.v_1 + R_2.v_2$$
(čteme: $v$ je definováno jako $R_1$ tečka $v_1$ nebo $R_2$ tečka $v_2$) nebo
$$v \stackrel{\text{def}}{=} \sum_{i \in I} R_i.v_i,$$
kde $I$ je nějaká množina (čteme: $v$ je definováno jako suma $R_i$ tečka $v_i$ pro $i \in I$).

V odstavci 3.2 jsme uvedli, jakým způsobem lze akční strukturu typu $\Delta$ notací CCS zadat.

**9.1.1 Příklad** Ať $\Delta = \{\text{in}_1, \text{in}_2, \text{in}_3, \text{out}_1, \text{out}_2, \text{out}_3\}$. Uvažujme o akční struktuře, zadané následovně:

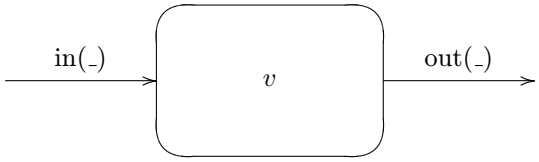
$$v \stackrel{\text{def}}{=} \text{in}_1.v_1 + \text{in}_2.v_2 + \text{in}_3.v_3 \tag{9.1}$$
$$v_1 \stackrel{\text{def}}{=} \text{out}_1.v \tag{9.2}$$
$$v_2 \stackrel{\text{def}}{=} \text{out}_2.v \tag{9.3}$$
$$v_3 \stackrel{\text{def}}{=} \text{out}_3.v \tag{9.4}$$

Obrázek této akční struktury vypadá takto:



Vrchol $v$ si můžeme představit jako jednomístný zásobník, který je schopen přijmout jednu z hodnot 1, 2, 3 a přejít do stavu $v_1$, $v_2$, $v_3$ (akce $\mathrm{in}_1$, $\mathrm{in}_2$, $\mathrm{in}_3$). Ze stavů $v_1$, $v_2$, $v_3$ pak zásobník přechází do počátečního stavu $v$ vydáním příslušné hodnoty (akce $\mathrm{out}_1$, $\mathrm{out}_2$, $\mathrm{out}_3$).

Označme $D = \{1, 2, 3\}$ a chápejme množinu $D$ jako množinu hodnot, které může přijímat a vysílat *proces* $v$ mající dva *komunikační porty*, vstupní $\mathrm{in}(_)$ a výstupní $\mathrm{out}(_)$ pro hodnoty $x \in D$:




Zatímní definice jazyka CCS nám dovoluje přepsat rovnice (9.1)–(9.4) jako

$$v \stackrel{\mathrm{def}}{=} \sum_{x \in D} \mathrm{in}_x.v_x \tag{9.5}$$

$$v_x \stackrel{\mathrm{def}}{=} \mathrm{out}_x.v, \quad x \in D \tag{9.6}$$

Kdybychom měli dovoleno akce parametrisovat, mohli bychom namísto rovnic (9.5) a (9.6) psát:

$$v \stackrel{\mathrm{def}}{=} \mathrm{in}(x).v_x, \quad x \in D \tag{9.7}$$

$$v_x \stackrel{\mathrm{def}}{=} \mathrm{out}(x).v, \quad x \in D \tag{9.8}$$

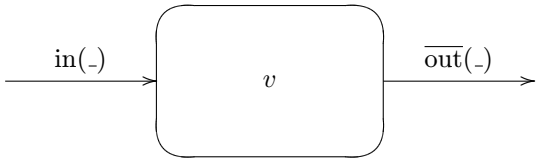
nebo dokonce

$$v \stackrel{\mathrm{def}}{=} \mathrm{in}(x).v(x), \quad x \in D \tag{9.9}$$

$$v(x) \stackrel{\mathrm{def}}{=} \mathrm{out}(x).v, \quad x \in D \tag{9.10}$$

v případě, že máme dovoleno parametrisovat i vrcholy.

Dále budeme chtít v jazyce CCS rozlišit „orientaci“ obou komunikačních portů procesu $v$: $\mathrm{in}(_)$ je vstupní port a $\mathrm{out}(_)$ je port výstupní. Toho lze dosáhnout například tím, že výstupní porty budeme vždy psát *s dekorací* $\overline{(_)}$, tj. obrázek procesu $v$ bude nyní vypadat takto:



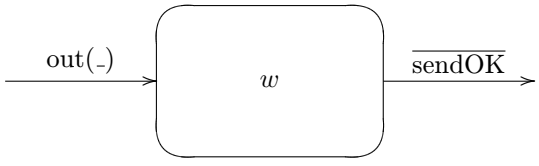

(9.11)

a rovnice CCS, které proces $v$ popisují, budou vypadat takto:

$$v \stackrel{\mathrm{def}}{=} \mathrm{in}(x).v(x), \quad x \in D \tag{9.12}$$

$$v(x) \stackrel{\mathrm{def}}{=} \overline{\mathrm{out}}(x).v, \quad x \in D \tag{9.13}$$

Značení akcí s dekoracemi má navíc tu výhodu, že budeme moci určit *partnery pro komunikaci*, například proces




může komunikovat s procesem 9.11 přes port $\mathrm{out}(_)$, protože proces $v$ má tento port označen jako výstupní a proces $w$ má tento port označen jako vstupní.

Všimněme si, že výstupní port sendOK procesu $w$ jsme zapsali způsobem, který naznačuje, že po něm *neposíláme* hodnoty z množiny $D$.

Úvahy z předchozího příkladu nyní shrneme v přesnou definici syntaxe jazyka CCS.

**9.1.2 Syntaxe CCS** Jazyk CCS dovoluje zkonstruovat *termy CCS* a *fráze CCS*.

1. Jako obvykle, $\Delta$ značí množinu jmen akcí. Jako $\overline{\Delta}$ označíme množinu jmen akcí tvaru $\overline{R}$, kde $R \in \Delta$. O akcích $R$ a $\overline{R}$ říkáme, že jsou *partnery pro komunikaci*. Z důvodu zjednodušení značení klademe $\overline{(\overline{R})} = R$ a definujeme obecně $\overline{A} = \{\overline{R} \mid R \in A\}$ pro libovolnou množinu akcí.

   Akcím z množiny $\Delta \cup \overline{\Delta}$ budeme říkat *pozorovatelné akce*. Jsou to ty akce, které se mohou případně zúčastnit komunikace. Dále zvolíme symbol $\tau \notin \Delta \cup \overline{\Delta}$ a tomuto symbolu budeme říkat *tichá akce*. Význam tiché akce bude patrný z následujících definic, budeme si ji představovat jako akci, která nám dává informaci o tom, že došlo ke komunikaci. Akce $\tau$ se však komunikace zúčastnit *nemůže*: její partner není definován.

   Množinu $\Delta \cup \overline{\Delta} \cup \{\tau\}$ označíme $\Omega$ a jejím prvkům budeme opět říkat akce, k nedorozumění nedojde. Typický prvek množiny $\Omega$ budeme značit $R$.

2. Budeme předpokládat, že je dána neprázdná množina $V$ — množina *jmen procesů*.

   Typické jméno procesu budeme značit $v$.

3. *Termy jazyka CCS nad množinami* $V$ *a* $\Omega$ jsou definovány následovně:

   (a) Každá jméno pro proces $v$ je term.

   (b) Je-li $R \in \Omega$ a je-li $t$ term, potom $(R.t)$ je term.
   Term $(R.t)$ čteme $R$ *tečka* $t$.

   (c) Je-li $D$ množina a je-li $t_d$ term pro každé $d \in D$, potom $(\sum_{d \in D} t_d)$ je term.

   Term $(\sum_{d \in D} t_d)$ čteme *suma* $t_d$ *pro* $d \in D$. Term $(\sum_{d \in D} t_d)$ pro $D = \emptyset$ označíme $\mathbf{0}$.

   (d) Jsou-li $t_1$ a $t_2$ termy, potom $(t_1 \mid t_2)$ je term.
   Term $(t_1 \mid t_2)$ čteme $t_1$ *spojeno s* $t_2$.

   (e) Je-li $t$ term a $A \subseteq \Delta \cup \overline{\Delta}$, potom $(t \setminus A)$ je term.
   Term $(t \setminus A)$ čteme *privatisace* $A$ *v* $t$. Povšimněme si, že v definici termu $(t \setminus A)$ smí $A$ být podmnožinou pouze množiny pozorovatelných akcí. Tichá akce $\tau$ tedy nesmí být privatisována. Obsahuje-li množina $A$ pouze jeden prvek, řekněme $R$, píšeme $(t \setminus R)$ namísto správného $(t \setminus \{R\})$.

   (f) Žádným jiným způsobem, než konečným použitím výše uvedených pravidel, term jazyka CCS vzniknout nemůže.

4. Množinu všech termů jazyka CCS nad množinami $V$ a $\Omega$ označíme $CCS(V, \Omega)$.

5. *Fráze jazyka CCS* nad $CCS(V, \Omega)$ je výraz $v \stackrel{\text{def}}{=} t$, kde $t$ je term a $v$ je jméno pro proces.

Jako vždy zavedeme konevenci o vnějších závvorkách, tj. píšeme například $t_1 \mid t_2$ namísto správného $(t_1 \mid t_2)$.
Termy a fráze jazyka CCS budeme interpretovat v jisté akční struktuře. Půjde o příklad *syntaktického modelu* — různé termy jsou interpretovány jako různé vrcholy.

**9.1.3 Sémantika CCS** Pro množinu termů $CCS(P, \Omega)$ definujeme akční strukturu $\mathbb{CCS}$ typu $\Omega$, jejíž vrcholy jsou termy jazyka CCS a akce jsou dány následujícími pravidly:

$$\frac{}{R.t \stackrel{R}{\to} t} \text{ (prefix)}, \qquad \text{je-li } R \in \Omega \tag{9.14}$$

$$\frac{t \stackrel{R}{\to} t'}{v \stackrel{R}{\to} t'} \ (\stackrel{\text{def}}{=}), \qquad \text{je-li } v \stackrel{\text{def}}{=} t \tag{9.15}$$

$$\frac{t_{d_0} \xrightarrow{R} t}{\sum_{d\in D} t_d \xrightarrow{R} t}\ (\textstyle\sum), \qquad \text{je-li } d_0 \in D \tag{9.16}$$

$$\frac{t_1 \xrightarrow{R} t_1'}{t_1 \mid t \xrightarrow{R} t_1' \mid t}\ (\text{NKl}) \tag{9.17}$$

$$\frac{t_2 \xrightarrow{R} t_2'}{t \mid t_2 \xrightarrow{R} t \mid t_2'}\ (\text{NKr}) \tag{9.18}$$

$$\frac{t_1 \xrightarrow{R} t_1' \quad t_2 \xrightarrow{\overline{R}} t_2'}{t_1 \mid t_2 \xrightarrow{\tau} t_1' \mid t_2'}\ (\text{PK}) \tag{9.19}$$

$$\frac{t \xrightarrow{R} t'}{t \setminus A \xrightarrow{R} t' \setminus A}\ (\text{Priv}), \qquad \text{jestliže } R \notin A \text{ a } \overline{R} \notin A \tag{9.20}$$

Jednotlivá pravidla nyní vysvětlíme:

Odvozovací pravidlo (9.14) je třeba číst jako axiom: na to, abychom odvodili akci $R.t \xrightarrow{R} t$, potřebujeme vědět pouze, že $R \in \Omega$.

Pravidlo (9.15) je odvozovací pravidlo: víme-li, že $v \stackrel{\text{def}}{=} t$ a víme-li, že $t \xrightarrow{R} t'$, pak odvodíme akci $v \xrightarrow{R} t'$.

Pravidlo (9.16) je také chápáno jako odvozovací pravidlo: víme-li, že $d_0 \in D$ a víme-li, že $t_{d_0} \xrightarrow{R} t$, pak odvodíme akci $\sum_{d\in D} t_d \xrightarrow{R} t$.

Intuitivní význam pravidel (9.14)– (9.16) byl vysvětlen v odstavci 3.2. Intuitivní význam dalších pravidel je:

Pravidlům (9.17) a (9.18) budeme říkat pravidla pro *neúspěšný pokus o komunikaci*. Například pravidlo (9.17) říká, že jestliže proces $t_1$ vykonává akci $R$ (a stává se tak procesem $t_1'$), potom spojení procesů $t_1$ a $t$ (čili proces $t_1 \mid t$) vykonává také akci $R$ a stává se procesem $t_1' \mid t$. Na pokus o komunikaci proces (přes port $R$) $t$ *nereaguje*, to zdůvodňuje termín *neúspěšný pokus*. Analogický význam má pravidlo (9.18).

Pravidlo (9.19) je pravidlem o *úspěšné komunikaci* (říkáme také: *perfektní komunikace*). Jestliže proces $t_1$ vykonává akci $R$ a proces $t_2$ vykonává partnerskou akci $\overline{R}$, potom proces $t_1 \mid t_2$ vykonává tichou akci $\tau$. Existence tiché akce z procesu $t_1 \mid t_2$ do procesu $t_1' \mid t_2'$ tedy potvrzuje, že došlo ke komunikaci (přes port $R$).

Pravidlu (9.20) říkáme pravidlo o *privatisaci akcí z množiny* $A$. Jak uvidíme v příkladu 9.1.6, jde o jistý způsob vynucení komunikace.

**9.1.4 Poznámka** Pravidla pro tvorbu termů jazyka CCS nám *nedovolila* zápis

$$\begin{aligned} v &\stackrel{\text{def}}{=} \text{in}(x).v(x), \quad x \in D \\ v(x) &\stackrel{\text{def}}{=} \text{out}(x).v, \quad x \in D \end{aligned}$$

používaný v příkladu 9.1.1. Tento zápis je však intuitivně velmi názorný a my jej budeme považovat za *syntaktickou zkratku* pro správný zápis

$$\begin{aligned} v &\stackrel{\text{def}}{=} \sum_{d\in D} \text{in}_d.v_d \\ v_d &\stackrel{\text{def}}{=} \text{out}_d.v, \quad d \in D \end{aligned}$$

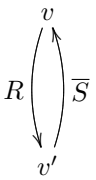
Poznamenejme, že podle definice má akční struktura $\mathbb{CCS}$, kterou jsme zavedli pravidly (9.14)–(9.20), *vždy* spočetně mnoho různých vrcholů. Vrcholy struktury $\mathbb{CCS}$ jsou totiž přesně termy jazyka CCS a protože jsme při tvorbě množiny termů $CCS(V, \Omega)$ předpokládali, že množina jmen pro procesy je neprázdná, musí být termů jazyka CCS spočetně mnoho.

**9.1.5 Příklad** Vývoj procesu můžeme graficky znázornit. Uvažujme o typu $\Delta = \{R, S\}$. Množina $\Omega$ je tedy pětiprvková množina $\{R, S, \overline{R}, \overline{S}, \tau\}$. Ukážeme nyní grafické vyjádření vývoje několika procesů.

Předpokládejme, že procesy $v$ a $w$ jsou v jazyce CCS definovány takto:

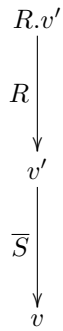
$$\begin{array}{rcl} v & \stackrel{\text{def}}{=} & R.v' \\ v' & \stackrel{\text{def}}{=} & \overline{S}.v \\ w & \stackrel{\text{def}}{=} & S.w' \\ w' & \stackrel{\text{def}}{=} & \overline{R}.w \end{array}$$

1. Vývoj procesu $v$ lze znázornit následovně:




Vývojem procesu $v$ tu rozumíme znázornění všech akcí, které lze pravidly (9.14)–(9.20) ve struktuře $\mathbb{CCS}$ odvodit. Termy, do kterých se dostaneme jedinou akcí z vrcholu $v$ bychom mohli nazvat *první generací vrcholů vzniklou z* $v$. Vývoj procesu $v$ však první generací nekončí: pro každý vrchol $t$ z první generace je zapotřebí sestavit první generací vrcholů vzniklou z $t$ a tak pokračovat. V našem případě se vývoj procesu ukončil po dvou krocích: vrchol $v$ je ve druhé generaci termů vzniklých z vrcholu $v$.

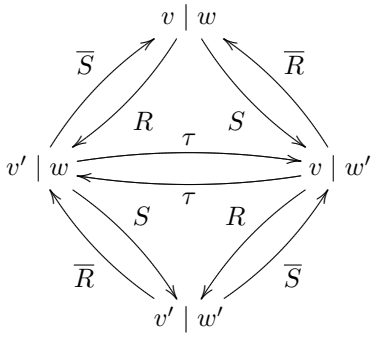
Všimněme si, že vývoj procesu $R.v'$ vypadá z počátku velmi podobně, jako vývoj procesu $v$:




Tento vývoj se *nezastaví* po dvou generacích: vrcholy $v$ a $R.v'$ jsou různé, protože jde o různé termy! Z vrcholu $v$ tedy musíme pokračovat dále akcí $R$ z $v$ do $v'$ a celkový vývoj procesu $R.v'$ můžeme znázornit následovně:

Intuitivní význam definice $v \stackrel{\text{def}}{=} R.v'$ však je ten, že proces $v$ je „stejný“ jako proces $R.v'$ — oba procesy by tedy měly mít stejný vývoj. Bude tomu tak, pokud na oba procesy budeme nahlížet optikou *silné bisimilarity* procesů. Těmto problémům se budeme věnovat v odstavci 9.2. K ztotožnění vývojů procesů $v$ a $R.v'$ se vrátíme v příkladu 9.2.1.

2. Znázorněme vývoj procesu $v \mid w$. Budeme již postupovat rychleji. Pravidla jazyka CCS nám dávají následující obrázek:




(9.21)

   Poznamenejme, že tento obrázek lze chápat i jako vývoj procesu $v' \mid w$. Tento fakt budeme potřebovat v příkladu 9.1.6.

3. Vývoj procesu $(v \mid w) \setminus \{S, R\}$ je nejjednodušší: pravidla jazyka CCS totiž pro tento proces nedovolují odvodit žádnou akci.

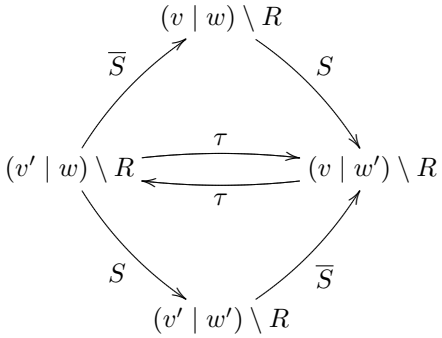
$$(v \mid w) \setminus \{S, R\}$$

Výše uvedeným grafům budeme říkat *evoluční diagram procesu*. Viz také cvičení 9.5.1.

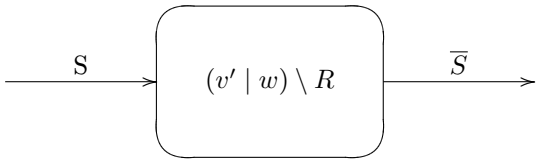
**9.1.6 Příklad** V tomto příkladu vysvětlíme pravidlo (9.20) o privatisaci akcí jako požadavek na vynucení komunikace. Použijeme definice procesů $v$, $v'$, $w$ a $w'$ z předchozího příkladu:

$$\begin{array}{rcl} v & \stackrel{\text{def}}{=} & R.v' \\ v' & \stackrel{\text{def}}{=} & \overline{S}.v \\ w & \stackrel{\text{def}}{=} & S.w' \\ w' & \stackrel{\text{def}}{=} & \overline{R}.w \end{array}$$

Nakresleme evoluční diagram procesu $(v' \mid w) \setminus R$:




Výše uvedený obrázek se liší od diagramu (9.21) (který můžeme chápat jako evoluční diagram procesu $v' \mid w$) tím, že odpadly neúspěšné pokusy o komunikaci přes port $R$. Navíc proces $(v' \mid w) \setminus R$ komunikační port $R$ „skryl“ před okolním světem: žádný jiný proces nemůže s procesem $(v' \mid w) \setminus R$ přes port $R$ komunikovat. Proces $(v' \mid w) \setminus R$ má pouze jeden vstupní port $S$ a pouze jeden výstupní port $S$:



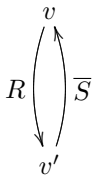
## 9.2 Silná bisimilarita procesů

V tomto odstavci zavedeme pojem rovnosti na termech jazyka CCS. Využijeme pojmů zig-zag relace na akčních strukturách z kapitoly 3. Začneme příkladem.

**9.2.1 Příklad** Připomeňme definice procesů $v$ a $v'$ z příkladu 9.1.5:

$$\begin{array}{lll} v & \stackrel{\text{def}}{=} & R.v' \\ v' & \stackrel{\text{def}}{=} & \overline{S}.v \end{array}$$

Jako evoluční diagram procesu $v$ jsme uvedli diagram




a jako evoluční diagram procesu $R.v'$ jsme v příkladu 9.1.5 sestrojili diagram

V tomto odstavci se budeme zabývat následujícím problémem:

> Výrazy $v$ a $R.v'$ jsou *různé* jako termy jazyka CCS, fráze $v \stackrel{\text{def}}{=} R.v'$ je však *ztotožňuje* jako procesy.

Musíme tedy zavést pojem, který nám umožní prohlásit, že výše uvedené evoluční diagramy znázorňují *stejné chování* procesů $v$ a $R.v'$.

Nabízí se následující myšlenka: chování procesů $v$ a $R.v'$ je stejné přesně tehdy, když v prvních generacích obou procesů dostaneme procesy, které mají stejné chování.

Pro odpověď na otázku, v jakém smyslu rovnost $v \stackrel{\text{def}}{=} R.v'$ procesy $v$ a $R.v'$ ztotožňuje, odkazujeme na cvičení 9.5.7.

Rovnost procesů (tj. stejnost chování), kterou hodláme zavést, bude tedy rovností v *sémantickém smyslu*: řekneme-li, že dva termy jsou si rovny, rozumíme tím, že se chovají stejně jako procesy. Abychom se vyhnuli přetížení slova „rovnost“, budeme pro pojem ztotožňování termů (procesů) používat obvyklý termín *silná bisimilarita*. V teorii procesů se ještě používá jiný pojem rovnosti: *bisimilarita* (také nazývaná *pozorovatelná ekvivalence*). Zavedení pojmu bisimilarity je však technicky náročné a my odkazujeme například na knihu [Mi89].

Podle intuice z příkladu 9.2.1 by tedy pojem silné bisimilarity měl mít následující vlastnost:

> Termy $t_1$ a $t_2$ silně bisimilární přesně tehdy, když v prvních generacích procesů $t_1$ a $t_2$ jsou procesy silně bisimilární. Tento požadavek je nutno vyjádřit přesněji: procesy v první generaci mohly z termů $t_1$ a $t_2$ vzniknout nejrůznějšími akcemi. Budeme chtít, aby v první generaci byly silně bisimilární přesně ty procesy, které z termů $t_1$ a $t_2$ vznikly akcemi stejného jména.

Výše uvedený požadavek nyní zformulujeme formálně. Fakt, že termy $t_1$ a $t_2$ jsou silně bisimilární, zapíšeme jako $t_1 \rightleftharpoons t_2$.

> Požadujeme, aby platilo $t_1 \rightleftharpoons t_2$ *právě tehdy*, když pro každou akci $R \in \Omega$ platí následující dva požadavky:

Je-li $t_1 \xrightarrow{R} t_1'$, potom existuje $t_2'$ tak, že $t_2 \xrightarrow{R} t_2'$ a $t_1' \rightleftharpoons t_2'$.
Je-li $t_2 \xrightarrow{R} t_2'$, potom existuje $t_1'$ tak, že $t_1 \xrightarrow{R} t_1'$ a $t_1' \rightleftharpoons t_2'$.

V předchozích dvou požadavcích rozpoznáváme podmínky (B) a (F) z definice zig-zag relace na akční struktuře $\mathbb{CCS}$ — viz definice 3.3.7. Povšimněme si však zásadního rozdílu: binární relace $C$ na $\mathbb{CCS}$ je zig-zag, pokud platí:

Jestliže $t_1\ C\ t_2$, *potom* pro každou akci $R \in \Omega$ platí následující dva požadavky:

Je-li $t_1 \xrightarrow{R} t_1'$, potom existuje $t_2'$ tak, že $t_2 \xrightarrow{R} t_2'$ a $t_1'\ C\ t_2'$.
Je-li $t_2 \xrightarrow{R} t_2'$, potom existuje $t_1'$ tak, že $t_1 \xrightarrow{R} t_1'$ a $t_1'\ C\ t_2'$.

**9.2.2 Příklad** Vyzkoušejme, jestli naše definice silné bisimilarity vystihuje stejnost chování procesů $v$ a $R.v'$ z příkladu 9.2.1.

Abychom ukázali, že procesy $v$ a $R.v'$ jsou silně bisimilární, potřebujeme ukázat, že proces $v'$ je silně bisimilární s procesem $v'$. Jiné procesy totiž v první generaci nejsou.

Všimněme si ale, že relace silné bisimilarity je jistě reflexivní: pro libovolný term $t$ jazyka CCS platí $t \rightleftharpoons t$.

Relace $\rightleftharpoons$ bude definována na speciálním případu akční struktury — na struktuře $\mathbb{CCS}$. Definujme nyní obecný pojem silné bisimilarity na (libovolné) akční struktuře, relace $\rightleftharpoons$ pak bude speciálním případem tohoto obecného pojmu.

**9.2.3 Definice** Řekneme, že binární relace $C$ na akční struktuře $\mathbb{V}$ je *silná bisimilarita*, pokud platí: $v\ C\ w$ právě tehdy, když pro každou akci $R$ platí následující dva požadavky:

(SF) Je-li $v \xrightarrow{R} v'$, potom existuje $w'$ tak, že $w \xrightarrow{R} w'$ a $v'\ C\ w'$.

(SB) Je-li $w \xrightarrow{R} w'$, potom existuje $v'$ tak, že $v \xrightarrow{R} v'$ a $v'\ C\ w'$.

Podmínky (SF) a (SB) jsou *tytéž* jako podmínky (F) a (B) z definice pojmu zig-zag relace — definice 3.3.7. Pojmy zig-zag relace a silné bisimilarity se však *liší* (viz cvičení 9.5.2), proto jsme zvolili odlišné označení těchto podmínek.

Připomeňme několik vlastností obecných zig-zag relací, které budeme v následujícím potřebovat:

**9.2.4 Lemma** *Předpokládejme, že $\mathbb{V}$ je libovolná akční struktura. Potom platí:*

1. *Identická relace na $\mathbb{V}$ je zig-zag relace.*
2. *Jsou-li $C_1$ a $C_2$ zig-zag relace na $\mathbb{V}$, potom i relace $C_1 \circ C_2$ je zig-zag relace na $\mathbb{V}$.*
3. *Je-li $C$ zig-zag relace na $\mathbb{V}$, potom i relace $C^{op}$ je zig-zag relace na $\mathbb{V}$.*
4. *Je-li $C_i$, $i \in I$ libovolný systém zig-zag relací na $\mathbb{V}$, potom i relace $C$, definovaná jako $\bigcup_{i \in I} C_i$, je zig-zag relace na $\mathbb{V}$.*

Důkaz. Vlastnosti 1. a 2. plynou z lemmatu 3.3.10, vlastnost 3. byla motivační vlastností pro zavedení zig-zag relací v odstavci 3.3. Zbývá tedy dokázat vlastnost 4.: předpokládejme, že $v\ C\ w$ a že $v \xrightarrow{R} v'$. Potom existuje $i \in I$ tak, že $v\ C_i\ w$. Protože $C_i$ je zig-zag relace, existuje $w'$ tak, že $w \xrightarrow{R} w'$ a platí $v'\ C_i\ w'$. Ukázali jsme, že platí $v'\ C\ w'$. Relace $C$ má vlastnost (F).

Vlastnost (B) relace $C$ se ukáže analogicky. ■

**9.2.5 Značení** Relaci $S$ na akční struktuře $\mathbb{V}$ definujeme rovností

$$S = \bigcup \{C \mid C \text{ je zig-zag relace na } \mathbb{V}\}.$$

**9.2.6 Lemma** *Pro relaci $S$ zavedenou v 9.2.5 platí:*

1. *Relace $S$ je zig-zag relace.*
2. *Relace $S$ je ekvivalence.*

Důkaz. Vlastnost 1. plyne okamžitě z tvrzení 4. lemmatu 9.2.4. Pro důkaz vlastnosti 2. ukážeme, že $S$ je reflexivní, symetrická a transitivní.

- Reflexivita relace $S$ plyne z toho, že identická relace na $\mathbb{V}$ je zig-zag (tvrzení 1. lemmatu 9.2.4).
- Symetrie relace $S$ plyne z toho, že $C^{op}$ je zig-zag relace pro libovolnou zig-zag relaci $C$ (tvrzení 3. lemmatu 9.2.4).
- Pro důkaz transitivity $S$ předpokládejme, že $v_1\ S\ v_2$ a $v_2\ S\ v_3$. Potom existují zig-zag relace $C_1$ a $C_2$ tak, že $v_1\ C_1\ v_2$ a $v_2\ C_2\ v_3$. Podle definice skládání relací platí $v_1\ C_1 \circ C_2\ v_3$. Relace $C_1 \circ C_2$ je ale zig-zag (tvrzení 2. lemmatu 9.2.4), tedy platí $v_1\ S\ v_3$.

■

Abychom ukázali, že relace $S$ definovaná v 9.2.5 je silná bisimilarita, využijeme Tarského větu o pevných bodech z odstavce 8.3. Binární relace $C$ na množině vrcholů $V$ akční struktury $\mathbb{V}$ je libovolná podmnožina kartézského součinu $V \times V$. Množina všech binárních relací na množině $V$ je tedy množina $exp(V \times V)$. Množinu $exp(V \times V)$ lze částečně uspořádat relací inkluse a výsledkem je úplný svaz $\langle exp(V \times V), \subseteq \rangle$. Označme tento úplný svaz jako $\mathbb{REL}(V)$.

**9.2.7 Značení** Funkci $f : \mathbb{REL}(V) \to \mathbb{REL}(V)$ definujeme takto: $f(C)$ je taková binární relace, která splňuje podmínku: $v\ f(C)\ w$ platí právě tehdy, když pro každou akci $R$ platí následující dva požadavky:

Je-li $v \xrightarrow{R} v'$, potom existuje $w'$ tak, že $w \xrightarrow{R} w'$ a $v'\ C\ w'$.

Je-li $w \xrightarrow{R} w'$, potom existuje $v'$ tak, že $v \xrightarrow{R} w'$ a $v'\ C\ w'$.

Funkce $f$, kterou jsme právě zavedli, má tyto zajímavé vlastnosti.

**9.2.8 Lemma** *Pro funkci $f$ definovanou v 9.2.7 platí:*

1. *$f : \mathbb{REL}(V) \to \mathbb{REL}(V)$ je monotónní.*
2. *Postfixní body funkce $f$ jsou přesně všechny zig-zag relace.*
3. *Je-li relace $C$ pevným bodem funkce $f$, pak $C$ je silná bisimilarita.*
4. *Relace $S$ definovaná rovvností 9.2.5 je největším pevným bodem funkce $f$, tj. je největší bisimilaritou.*

Důkaz. Platnost tvrzení 1. plyne přímo z definice funkce $f$.

Tvrzení 2.: fakt, že relace $C$ je postfixním bodem funkce $f$ znamená přesně tolik, že platí $C \subseteq f(C)$. Jestliže tedy pro $v$ a $w$ platí $v\ C\ w$, potom platí $v\ f(C)\ w$. Podle definice zobrazení $f$ to znamená, že jsou splněny požadavky:

Je-li $v \xrightarrow{R} v'$, potom existuje $w'$ tak, že $w \xrightarrow{R} w'$ a $v'\ C\ w'$.

Je-li $w \xrightarrow{R} w'$, potom existuje $v'$ tak, že $v \xrightarrow{R} v'$ a $v'\ C\ w'$.

Ukázali jsme, že každý postfixní bod funkce $f$ je zig-zag relace.

Obráceně: je-li $C$ zig-zag relace a platí-li $v\ C\ w$, potom jsou (z definice zig-zag relace) splněny oba výše uvedené požadavky. To ale znamená, že platí $v\ f(C)\ w$ podle definice zobrazení $f$. Ukázali jsme, že pro každou zig-zag relaci $C$ platí inkluse $C \subseteq f(C)$, čili $C$ je postfixním bodem funkce $f$.

Tvrzení 3. plyne z toho, že relace $C$ je pevným bodem $f$, když platí $C = f(C)$.

Tvrzení 4. plyne okamžitě z Tarského věty 8.3.1 a tvrzení 2. tohoto lemmatu: největší pevný bod $f$ je supremum postfixních bodů. Suprema ve svazu $\mathbb{REL}(V)$ jsou přesně množinová sjednocení. ■

Vraťme se nyní k akční struktuře $\mathbb{CCS}$ termů jazyka CCS. Z lemmatu 9.2.8 víme, že definujeme-li binární relaci $\rightleftharpoons$ předpisem

$$\rightleftharpoons = \bigcup \{C \mid C \text{ je zig-zag relace na } \mathbb{CCS}\}, \tag{9.22}$$

dostaneme silnou bisimilaritu na akční struktuře $\mathbb{CCS}$. Relace $\rightleftharpoons$ nám umožní algebraické úpravy termů jazyka CCS. Základní algebraické úpravy shrneme do následujících tří vět. Důkazy těchto vět spočívají na jednoduché myšlence: abychom pro dva termy $t_1$ a $t_2$ ukázali, že platí $t_1 \rightleftharpoons t_2$, stačí nalézt zig-zag relaci $C$ takovou, že $t_1\ C\ t_2$. Z několika případů bude patrný obecný postup, jak takovou relaci $C$ nalézt, a proto pro podrobné důkazy následujících vět odkazujeme na cvičení.

**9.2.9 Věta (Polosvazové vlastnosti $+$)** *Předpokládejme, že $t_1$, $t_2$ a $t_3$ jsou termy jazyka CCS. Potom platí:*

1. $t_1 + t_1 \rightleftharpoons t_1$.
2. $t_1 + t_2 \rightleftharpoons t_2 + t_1$.
3. $t_1 + (t_2 + t_3) \rightleftharpoons (t_1 + t_2) + t_3$.
4. $t_1 + \mathbf{0} \rightleftharpoons t_1$. *(Připomínáme, že* $\mathbf{0}$ *je zkratka pro term* $\sum_{d \in D} t_d$, *kde* $D = \emptyset$.*)*

Důkaz. Ukážeme tvrzení 3., pro důkaz ostatních tvrzení odkazujeme na cvičení 9.5.3.

Stačí ukázat, že existuje zig-zag relace $C$ taková, že platí $t_1 + (t_2 + t_3)\ C\ (t_1 + t_2) + t_3$. Binární relaci $C$ definujeme takto:

> $v\ C\ w$ právě tehdy, když buď existují termy $v_1$, $v_2$, $v_3$ tak, že $v = v_1 + (v_2 + v_3)$ a $w = (v_1 + v_2) + v_3$ nebo $v = w$.

Podle definice relace $C$ zřejmě platí $t_1 + (t_2 + t_3)\ C\ (t_1 + t_2) + t_3$. Podle definice relace $\rightleftharpoons$ stačí ukázat, že relace $C$ je zig-zag. Ukážeme tedy, že $C$ má vlastnosti (B) a (F).

- Vlastnost (B): předpokládejme, že platí $v\ C\ w$ a $w \xrightarrow{R} w'$. Podle definice relace $C$ existují termy $v_1$, $v_2$, $v_3$ takové, že $v = v_1 + (v_2 + v_3)$ a $w = (v_1 + v_2) + v_3$.

  Definujme $v' = w'$. Podle definice relace $C$ platí $v'\ C\ w'$. Zbývá ukázat existenci akce $v \xrightarrow{R} v'$.

  Akce $w \xrightarrow{R} w'$ musela vzniknout pomocí pravidel $(\sum)$ z jedné z následujících akcí:

  $$v_1 \xrightarrow{R} w' \quad \text{nebo} \quad v_2 \xrightarrow{R} w' \quad \text{nebo} \quad v_3 \xrightarrow{R} w'.$$

  Probereme pouze první možnost, ostatní jsou analogické.

  Jestliže $v_1 \xrightarrow{R} w'$, potom použitím pravidla $(\sum)$ je $v_1 + (v_2 + v_3) \xrightarrow{R} w'$. Protože $v' = w'$, je důkaz u konce.

- Vlastnost (F) se ukáže analogicky jako vlastnost (B).

■

**9.2.10 Věta (Polosvazové vlastnosti $\mid$)** *Předpokládejme, že $t_1$, $t_2$ a $t_3$ jsou termy jazyka CCS. Potom platí:*

1. $t_1 \mid t_2 \rightleftharpoons t_2 \mid t_1$.
2. $t_1 \mid (t_2 \mid t_3) \rightleftharpoons (t_1 \mid t_2) \mid t_3$.
3. $t_1 \mid \mathbf{0} \rightleftharpoons t_1$. *(Připomínáme, že* $\mathbf{0}$ *je zkratka pro term* $\sum_{d \in D} t_d$, *kde* $D = \emptyset$.*)*

Důkaz. Naznačíme, jak ukázat platnost tvrzení 2., pro důkaz tvrzení 1. a 3. odkazujeme na cvičení 9.5.3.

Definujte binární relaci $C$ následovně:

$v\ C\ w$ právě tehdy, když existují termy $v_1$, $v_2$, $v_3$ tak, že $v = v_1 \mid (v_2 \mid v_3)$ a $w = (v_1 \mid v_2) \mid v_3$.

Protože zřejmě platí $t_1 \mid (t_2 \mid t_3)\ C\ (t_1 \mid t_2) \mid t_3$, stačí ukázat, že relace $C$ je zig-zag — viz cvičení 9.5.3. ■

**9.2.11 Věta (Substituční vlastnosti relace $\rightleftharpoons$)** *Předpokládejme, že $t_1$ a $t_2$ jsou termy jazyka CCS takové, že $t_1 \rightleftharpoons t_2$. Potom platí:*

1. *$R.t_1 \rightleftharpoons R.t_2$ pro libovolnou akci $R$.*
2. *$t_1 + t \rightleftharpoons t_2 + t$ pro libovolný term $t$.*
3. *$t_1 \mid t \rightleftharpoons t_2 \mid t$ pro libovolný term $t$.*
4. *$t_1 \setminus A \rightleftharpoons t_2 \setminus A$ pro libovolnou množinu $A$ pozorovatelných akcí.*

Důkaz. Viz cvičení 9.5.6. ■

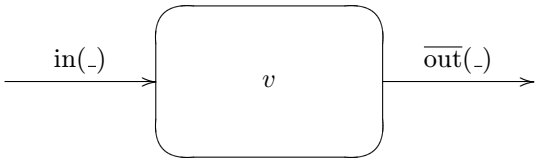
## 9.3 Přejmenovávání komunikačních portů

V tomto odstavci vysvětlíme konstrukci přejmenovávání komunikačních portů. Tím ukončíme definici jádra jazyka CCS. Existuje celá řada variant jazyka CCS — zde odkazujeme například na [Hoa85] nebo na [He88]. Začneme motivačním příkladem.

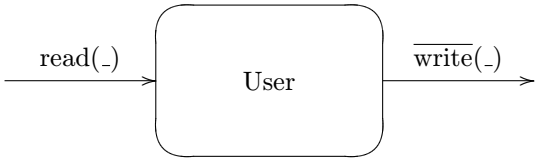
**9.3.1 Příklad** Připomeňme jednomístný zásobník z příkladu 9.1.1:

$$\begin{aligned} v &\stackrel{\text{def}}{=} \text{in}(x).v(x), \quad x \in D \\ v(x) &\stackrel{\text{def}}{=} \text{out}(x).v, \quad x \in D \end{aligned}$$

a jeho grafické znázornění




Definujme nyní proces User takto:




Oba procesy $v$ a User spolu sice můžeme spojit v proces $v \mid$ User, jakýkoli pokus o komunikaci však bude neúspěšný — nejsou zde žádní partneři pro potencionální komunikaci.

Kdybychom ale vstupní port procesu User *přejmenovali* na out(_), mohly by procesy $v$ a takto přejmenovaný proces User přes tento port úspěšně komunikovat. Označme přejmenovaný proces jako Reader. Navíc cítíme, že podstatu procesu User bychom přejmenováním nijak zvlášť nenarušili, původní proces User a proces Reader jsou „stejné“ v tom smyslu, že oba jsou (stejně jako proces $v$) pouze instancemi procesu s jedním vstupním a jedním výstupním portem.

Myšlenku přejmenování komunikačních portů vystihuje pojem *funkce přejmenování*, zavedený v následující definici.

**9.3.2 Definice** Funkci $f : \Omega \to \Omega$ říkáme *funkce přejmenování*, pokud splňuje následující tři podmínky:

1. Funkce $f$ *respektuje dekorace*, tj. platí $f(\overline{R}) = \overline{f(R)}$ pro každou pozorovatelnou akci $R$.
2. Funkce $f$ *respektuje tichou akci*, tj. platí rovnost $f(\tau) = \tau$.
3. Funkce $f$ *nenarušuje přenos dat*, tj. rovnost $f(R(x)) = (f(R))(x)$ platí pro každý parametrisovaný systém pozorovatelných akcí $\{R(x) \mid x \in D\}$.

Důkaz následujícího lemmatu přenecháváme jako cvičení 9.5.10:

**9.3.3 Lemma** *Identické zobrazení $1_\Omega : \Omega \to \Omega$ je funkce přejmenování. Složením dvou funkcí přejmenování vznikne opět funkce přejmenování.*

Funkce přejmenování dovolují rozšířit definici termů jazyka CCS o term $t[f]$, kde $t$ je term a $f$ je funkce přejmenování. Termy tohoto tvaru přidají do akční struktury akce vytvořené následujícím odvozovacím pravidlem:

$$\frac{t \xrightarrow{R} t'}{t[f] \xrightarrow{f(R)} t'[f]}\ (f), \qquad \text{jestliže } f \text{ je funkce přejmenování} \tag{9.23}$$

Základní vlastnosti konstrukce $t[f]$ lze shrnout do následující věty:

**9.3.4 Věta** *Pro libovolný term $t$ platí:*

1. $t[1_\Omega] \rightleftharpoons t$.
2. $(t[f])[g] \rightleftharpoons t[g \circ f]$ *pro libovolné funkce přejmenování $f$ a $g$.*

Důkaz. Viz cvičení 9.5.11. ■

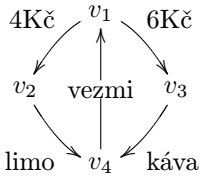
## 9.4 Uzavřené procesy

Fráze jazyka CCS jsme zavedli jako nástroj pro definování procesů. Definujeme-li však například proces $v$ jako term $R.v'$ (tj. napíšeme-li $v \stackrel{\text{def}}{=} R.v'$), nemůžeme o chování procesu $v$ říci nic více, než že v akční struktuře $\mathbb{CCS}$ lze odvodit akci $v \xrightarrow{R} v'$ (a samozřejmě také lze odvodit akci $R.v' \xrightarrow{R} v'$). Definice procesu $v$ není úplná v tom smyslu, že jsme nedefinovali chování procesu $v'$.

Definice jednoduchého automatu na kávu a limonádu ze cvičení 3.4.3 však výše uvedený nedostatek neměla; definovali jsme zde *každé* jméno procesu:

$$\begin{aligned} v_1 &\stackrel{\text{def}}{=} 4\text{Kč}.v_2 + 6\text{Kč}.v_3 \\ v_2 &\stackrel{\text{def}}{=} \text{limo}.v_4 \\ v_3 &\stackrel{\text{def}}{=} \text{káva}.v_4 \\ v_4 &\stackrel{\text{def}}{=} \text{vezmi}.v_1 \end{aligned}$$

Chování procesu $v_1$ jsme pak znázornili diagramem




Procesům, které jsou definovány podobně jako jednoduchý automat na kávu a limonádu, budeme říkat uzavřené procesy:

**9.4.1 Definice** *Uzavřený proces* je soutava frází jazyka CCS tvaru

$$v_i \stackrel{\text{def}}{=} t_i, \quad i \in I$$

kde $I$ je neprázdná množina a jména procesů, která se vyskytují v termech $t_i$ jsou prvky množiny $\{v_i \mid i \in I\}$.

Jestliže je množina $I$ konečná, mluvíme o *konečném uzavřeném procesu*.

Je zřejmé, že každý konečný uzavřený proces lze nahradit jedinou definiční rovnicí. Tak jsme například mohli zaměnit systém definic

$$\begin{aligned} v_1 &\stackrel{\text{def}}{=} 4\text{Kč}.v_2 + 6\text{Kč}.v_3 \\ v_2 &\stackrel{\text{def}}{=} \text{limo}.v_4 \\ v_3 &\stackrel{\text{def}}{=} \text{káva}.v_4 \\ v_4 &\stackrel{\text{def}}{=} \text{vezmi}.v_1 \end{aligned}$$

jedinou rovnicí

$$w_1 \stackrel{\text{def}}{=} 4\text{Kč.limo.vezmi}.w_1 + 6\text{Kč.káva.vezmi}.w_1$$

Procesy $v_1$ a $w_1$ jsou silně bisimilární (viz cvičení 9.5.13), jejich chování je tedy stejné.

Uveďme netriviální příklad konečného uzavřeného procesu:

**9.4.2 Příklad** Následující příklad železničního přejezdu pochází z [BS90].

$$\begin{aligned} \text{Rail} &\stackrel{\text{def}}{=} \text{train.green.}\overline{\text{tcross}}.\overline{\text{red}}.\text{Rail} \\ \text{Road} &\stackrel{\text{def}}{=} \text{car.up.}\overline{\text{ccross}}.\overline{\text{down}}.\text{Road} \\ \text{Signal} &\stackrel{\text{def}}{=} \overline{\text{green}}.\text{red.Signal} + \overline{\text{up}}.\text{down.Signal} \\ \\ \text{Crossing} &\stackrel{\text{def}}{=} (\text{Rail} \mid \text{Road} \mid \text{Signal}) \setminus \{\text{green}, \text{red}, \text{up}, \text{down}\} \end{aligned}$$

V procesu Crossing spolu komunikují tři procesy: Rail, Road a Signal. Proberme jejich intuitivní význam:

1. Proces Rail (koleje) má dva vstupní porty: port train (přijíždí vlak) a port green (dotaz na zelenou na semaforu) a dva výstupní porty: port $\overline{\text{tcross}}$ (vlak přejíždí přejezd) a port $\overline{\text{red}}$ (nastavení červené na semaforu).

2. Proces Road (silnice) má dva vstupní porty: port car (přijíždí auto) a port up (dotaz na zvižené závory) a dva výstupní porty: port $\overline{\text{ccross}}$ (auto přejíždí přejezd) a port $\overline{\text{down}}$ (spuštění závory).

3. Proces Signal (signalisační zařízení) má dva vstupní porty: port red (svítí červená) a port down (závory jsou spuštěné) a dva výstupní porty: port $\overline{\text{green}}$ (nastavení zelené na semaforu) a port $\overline{\text{up}}$ (vytažení závor).

Povšimněme si, že díky větě 9.2.10 smíme psát (Rail | Road | Signal) namísto syntakticky správného zápisu (Rail | Road) | Signal nebo Rail | (Road | Signal). Viz také cvičení 9.5.14.

## 9.5 Cvičení

**9.5.1 Cvičení** Dejte přesnou definici evolučního diagramu procesu. Návod: inspirujte se pojmem *komponenta souvislosti grafu*.

**9.5.2 Cvičení** Dejte příklad akční struktury a zig-zag relace, která není silnou bisimilaritou.

**9.5.3 Cvičení** Dokončete důkazy vět 9.2.9 a 9.2.10.

**9.5.4 Cvičení** Trojice $\langle X, *, e\rangle$, kde $*$ je binární operace na množině $X$ a $e \in X$ je *polosvaz s neutrálním prvkem*, když pro všechny prvky $x, y, z$ v $X$ platí:

1. $x * x = x$.
2. $x * y = y * x$.
3. $x * (y * z) = (x * y) * z$.
4. $x * e = x$.

Vysvětlete, proč jsme věty 9.2.9 a 9.2.10 nazvali větami o polosvazových vlastnostech + a $\mid$.

**9.5.5 Cvičení** Ukažte, že obecně neplatí $t_1 \setminus A \rightleftharpoons t_1$.
Nalezněte postačující podmínku pro to, aby následující relace $C$ byla zig-zag:

$v\ C\ w$ právě tehdy, když existuje term $t$ tak, že $v = t \setminus A$ a $w = t$.

Odvoďte z této postačující podmínky postačující podmínku pro platnost vztahu $t_1 \setminus A \rightleftharpoons t_1$.

**9.5.6 Cvičení** Dokažte větu 9.2.11.

**9.5.7 Cvičení** Ukažte, že relace $C$ definovaná následovně

$u\ C\ w$ právě tehdy, když buď $u$ je jméno pro proces a platí $u \stackrel{\text{def}}{=} w$ nebo $u = w$

je zig-zag a vyvoďte z tohoto faktu platnost tvrzení:

Jestliže $v \stackrel{\text{def}}{=} t$, potom $v \rightleftharpoons t$.

**9.5.8 Cvičení** Ukažte, že pro libovolné termy $t_1$, $t_2$, libovolnou množinu $A$ pozorovatelných akcí a libovolnou pozorovatelnou akci $R$ platí:

1. $t_1 + t_2 \setminus A \rightleftharpoons t_1 \setminus A + t_2 \setminus A$.
2. $(R.t_1) \setminus A \rightleftharpoons \mathbf{0}$, jestliže $R \in A \cup \overline{A}$. (Připomínáme, že $\mathbf{0}$ je zkratka pro term $\sum_{d \in D} t_d$, kde $D = \emptyset$.)
3. $(R.t_1) \setminus A \rightleftharpoons R.(t_1 \setminus A)$, jestliže $R \notin A \cup \overline{A}$.

**9.5.9 Cvičení** Rozhodněte, zda pro libovolné termy $t_1$, $t_2$ a $t_3$ a libovolnou pozorovatelnou akci $R$ platí:

1. $t_1 + \tau.(t_1 + t_2) \rightleftharpoons \tau.(t_1 + t_2)$.
2. $R.(t_1 + \tau.\tau.t_1) \rightleftharpoons R.t_1$.
3. $\tau.(t_1 + R.(t_2 + \tau.t_3)) \rightleftharpoons \tau.(t_1 + R.(t_2 + \tau.t_3)) + R.t_3$.

**9.5.10 Cvičení** Dokažte lemma 9.3.3.

**9.5.11 Cvičení** Dokažte větu 9.3.4.

**9.5.12 Cvičení** Rozhodněte, jaký vztah musí mít funkce přejmenování $f$ a $g$ tak, aby pro daný term $t$ platil vztah $t[f] \rightleftharpoons t[g]$.

**9.5.13 Cvičení** Ukažte, že procesy $v_1$ a $w_1$ jsou silně bisimilární, kde

$$\begin{array}{lcl} v_1 & \stackrel{\text{def}}{=} & 4\text{Kč}.v_2 + 6\text{Kč}.v_3 \\ v_2 & \stackrel{\text{def}}{=} & \text{limo}.v_4 \\ v_3 & \stackrel{\text{def}}{=} & \text{káva}.v_4 \\ v_4 & \stackrel{\text{def}}{=} & \text{vezmi}.v_1 \end{array}$$

$$w_1 \stackrel{\text{def}}{=} 4\text{Kč.limo.vezmi}.w_1 + 6\text{Kč.káva.vezmi}.w_1$$

**9.5.14 Cvičení** Zakreslete evoluční diagram uzavřeného procesu Crossing z příkladu 9.4.2.

**9.5.15 Cvičení** Následující příklad jednoduchého komunikačního protokolu pochází z [Wa87]. Předpokládejme, že je dána množina dat $D$, jejíž prvky posílá proces Sender prostřednictvím procesu Medium procesu Receiver a že $x \in D$. Komunikační protokol je proces Protocol.

$$\begin{array}{lcl} \text{Sender} & \stackrel{\text{def}}{=} & \text{in}(x).\overline{\text{sm}}(x).\text{Send1}(x) \\ \text{Send1}(x) & \stackrel{\text{def}}{=} & \text{ms}.\overline{\text{sm}}(x).\text{Send1}(x) + \text{ok.Sender} \\ \text{Medium} & \stackrel{\text{def}}{=} & \text{sm}(y).\text{Med1}(y) \\ \text{Med1}(y) & \stackrel{\text{def}}{=} & \overline{\text{mr}}(y).\text{Medium} + \tau.\overline{\text{ms}}.\text{Medium} \\ \text{Receiver} & \stackrel{\text{def}}{=} & \text{mr}(x).\overline{\text{out}}(x).\overline{\text{ok}}.\text{Receiver} \end{array}$$

$$\text{Protocol} \stackrel{\text{def}}{=} (\text{Sender} \mid \text{Medium} \mid \text{Receiver}) \setminus \{\text{sm}, \text{ms}, \text{mr}, \text{ok}\}$$

Vysvětlete práci konečného uzavřeného procesu Protocol.

# Literatura


[BaŠt86] Balcar B., Štěpánek P., *Teorie množin*, Academia, Praha 1986

[DePo97] Demlová M., Pondělíček B., *Matematická logika*, skriptum FEL ČVUT, Praha 1997

[BP89] Blok W. J., Pigozzi D.: *Algebraizable Logics*, Memoirs of the Amer. Math. Soc., No. 396, AMS, Providence 1989

[Bo94] Borceux F., *Handbook of Categorical Algebra*, Cambridge University Press, Cambridge 1994, (ve třech dílech)

[BS90] Bradfield J., Stirling C., *Verifying Temporal Properties of Processes*, in: *Lecture Notes in Computer Science 458*, 115–125

[DaP90] Davey B. A., Priestley H. A., *Introduction to Lattices and Order*, Cambridge University Press, Cambridge 1990

[Fi93] Fitting M., *Basic Modal Logic*, in: *Handbook of Logic in Computer Science*, Volume I: Logical Foundations, S. Abramsky, D. M. Gabbay and T. S. E. Maibaum (eds.), Oxford Science Publications, Clarendon Press, Oxford 1993, 368–448

[GaGu84] Gabbay D., Guenthner F., (eds.) *Handbook of Philosophical Logic*, Volume II: Extensions of Classical Logic, Kluwer Academic Publishers, Dordrecht 1984

[He88] Hennessy M., *Algebraic Theory of Processes*, MIT Press, Massachusetts 1985

[Hoa85] Milner R., *Communication Sequential Processes*, Prentice Hall International Series in Computer Science, 1985

[Che80] Chellas B. F., *Modal Logic: An Introduction*, Cambridge University Press, Cambridge, 1980

[Mi89] Milner R., *Communication and Concurrency*, Prentice Hall International Series in Computer Science, 1989

[RS63] Rasiowa H., Sikorski R.: *The Mathematics of Metamathematics*, PWN, Warszawa 1963

[RiVe97] Richta K., Velebil J., *Sémantika programovacích jazyků*, skriptum UK, Praha 1997

[St92] Stirling C., *Modal and Temporal Logics*, in: *Handbook of Logic in Computer Science*, Volume II: Computational Structures, S. Abramsky, D. M. Gabbay and T. S. E. Maibaum (eds.), Oxford Science Publications, Clarendon Press, Oxford 1992, 478–563

[Wa87] Walker D., *Introduction to a Calculus of Communication Systems*, Technical Report ECS-LFCS-87-22, Dept. of Computer Science, Edinburgh University

[We92] Wechler W., *Universal Algebra for Computer Scientists*, EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1992

# Rejstřík


$\alpha$-konverse, 82

abstraktní svaz, 55
akční struktura, 19
algebraická věta o korektnosti systému **K**, 63
algebraická věta o úplnosti systému **K**, 64
antisymetrická relace, 16
aritmetické výrazy, 44

Backusova-Naurova forma, 27
bijektivní zobrazení, 13
BNF, 27
Booleova algebra, 55
Booleovské výrazy, 44
box, 27

částečně uspořádaná množina, 54

deterministická akční struktura, 52
diamant, 27
disjunkce, 27
distributivní svaz, 55
dolní odhad, 54
duální zobrazení, 77

ekvivalence, 27
evoluční diagram procesu, 97
extense boxu, 57, 73
extense formule, 55
extense spojek, 74

faktorový model, 31
faktorový objekt akční struktury, 25
filtr, 66
formule parciální korektnosti, 46
fráze jazyka CCS, 94
funkce přejmenování, 103

Hennessyho-Milnerova modální logika, 27
Hoarova logika, 44
homomorfismus grafů, 14
horní odhad, 55

identické zobrazení, 13
identifikátory, 44
implikace, 27
infimum, 54
injektivní zobrazení, 13
ireflexivita, 41
isomorfismus grafů, 14
isomorfismus modelů, 31

jazyk predikátové logiky, 78
jazyk `while` programů, 44
jméno akce, 19
jméno procesu, 94

komplement, 55
komplementární svaz, 55
konečná množina, 13
konečné aproximace
    pevných bodů, 86
    sentencí modálního $\mu$-kalkulu, 87
konečný uzavřený proces, 104
konjunkce, 27
kontext pro proměnné, 79
koprodukt akčních struktur, 26
koprodukt grafů, 15
kvasiuspořádání, 16

Lindenbaumova-Tarského algebra, 63

modální $\mu$-model, 83
modální algebra, 58
modální model, 29
modální tautologie, 35
modálně definovatelná vlastnost, 39
monotónní zobrazení, 54

negace, 27
nejslabší liberální prepodmínka, 48, 69
notace CCS, 20

ohodnocení
    v Booleově algebře, 58
    v modální algebře, 61
orientovaný graf, 14

pevný bod, 72
podgraf, 14
podmodel, 31
podobjekt akční struktury, 25
podstruktura akční struktury, 25
polosvaz s neutrálním prvkem, 105
poset, 54
positivní forma formule, 81

positivní výskyt proměnné, 81
postfixní bod, 72
postpodmínka, 46
pozorovatelná akce, 94
prázdná množina, 13
pravdivost
    modální formule, 29
    sentence modálního $\mu$-kalkulu, 84
pravidlo Modus Ponens, 36
pravidlo zavedení boxu, 36
prefixní bod, 72
prepodmínka, 46
proces, 23
programy, 44
příkazy, 44
pullback modelu, 32

reflexivní a transitivní uzávěr relace, 39
reflexivní relace, 16
relace, 13
relace ekvivalence, 16
rozklad množiny, 16
rozšíření modelu, 32

sentence, 83
sémantika modálního $\mu$-kalkulu, 83
silná bisimilarita, 99
silný morfismus, 22
similarita, 16
součin grafů, 14
souvislá relace, 43
spočetná množina, 13
supremum, 55
surjektivní zobrazení, 13
svaz, 55
svaz s největším a nejmenším prvkem, 55
symetrická relace, 16
syntaktická složitost formule, 28
syntaktický model CCS, 94
syntaktický strom, 28
syntaxe modálního $\mu$-kalkulu, 82
systém **K**, 35
systém **T**, 47

tablo, 48
Tarského věta o pevném bodu, 72
term jazyka CCS, 94
tichá akce, 94
transitivní relace, 16
třída ekvivalence, 16

ultrafiltr v Booleově algebře, 60
uspořádání, 16
uzavřený proces, 104

úplná Booleova algebra, 76
úplný podgraf, 14
úplný svaz, 55
úspěšné tablo, 48

valuace, 29
věta o korektnosti systému **K**, 36
věta o korektnosti systému **T**, 49
věta o korespondenci, 38
věta o úplnosti systému **K**, 36, 64
věta o úplnosti systému **T**, 50
vlastnost
    Liveness, 84
    Safety, 84
volný monoid, 33
vynucení
    modální formule, 29
    sentence modálního $\mu$-kalkulu, 83

zig-zag isomorfismus, 24
zig-zag morfismus, 23
zig-zag morfismus modelů, 30
zig-zag relace, 23
zig-zag relace mezi modely, 30
zobecněná Churchova-Rosserova
    vlastnost, 38, 67
zobrazení, 13
Zornovo lemma, 66