Temporální logika reaktivních a konkurentních systémů

Patrik Halfar

Vysoké učení technické v Brně, Fakulta informačních technologií
Božetěchova 2, 612 66 Brno, CZ
www.fit.vutbr.cz/~ihalfar

Dílna moderní teoretické informatiky 2010, Brno

Obsah

Obsah

Obsah

- **Úvod do temporální logiky**
  - Stavová formule
  - Temporální formule
  - Dokazovací systém
- **Vlastnosti programů**
  - Přechodový systém
  - Lokální jazyk
  - Klasifikace vlastností
- **Příklad**

Prerekvizity

Výroková logika

- Logické spojky
- Normální formy
- Demorganovy zákony, distributivita
- Axiomy
- Odvozovací pravidla

Predikátová logika 1. řádu

- Kvantifikátory
- Axiomy
- Odvozovací pravidla

Definice: Stavová formule I

Slovník $\mathcal{V}$

Spočetná množina typovaných proměnných, které mohou nést data, nebo řídící informace.

Rozlišují se proměnné

flexibilní hodnota se mění mezi stavy výpočtu,

pevná konstantní pro všechny stavy výpočtu.

Další symboly

- Funkce
- Predikáty

Dále nechť je uvažováno $V \in \mathcal{V}$ jako podmnožina slovníku.

Definice: Stavová formule II

Výraz nad V

- Každá proměnná $x \in V$ je výraz nad V.
- Jsou-li $e_1, \ldots, e_m$ výrazy nad V a f je funkce arity m ($m \geq 0$), pak také $f(e_1, \ldots, e_m)$ je také výraz nad V.

Atomická formule nad V

- Každá proměnná $x \in V$, booleovského typu (tvrzení) je atomickou formulí nad V.
- Je-li P predikátem arity m ($m \geq 0$) a $e_1, \ldots, e_m$ jsou výrazy nad V, pak také $P(e_1, \ldots, e_m)$ je atomickou formulí nad V.

Definice: Stavová formule III

Booleovská formule nad V

- Každá atomická formule nad V je booleovskou formulí.
- Jsou-li p a q booleovské formule nad V, pak také
  $\neg p \qquad p \wedge q \qquad p \vee q \qquad p \rightarrow q \qquad p \leftrightarrow q$
  jsou booleovské formule nad V.

Stavová formule (výrok) nad V

- Každá booleovská formule nad V je výrokem nad V.
- Je-li p výrokem nad V, pak také
  $\exists u : p \qquad \forall u : p, \qquad u \in V$
  jsou výroky nad V.
- Jsou-li p a q výroky nad V, pak také
  $\neg p \qquad p \wedge q \qquad p \vee q \qquad p \rightarrow q \qquad p \leftrightarrow q$
  jsou výroky nad V.

Sémantika stavových formulí I

Modely nad V

Stav nad V je realizací, která každé proměnné $u \in V$ přiřazuje hodnotu z odpovídajícího typu, značí se

$$s[u].$$

Model σ nad V je nekonečným sledem tvaru

$$\sigma : s_0, s_1, s_2, \ldots,$$

kde každé s_i je stavem nad V.

Sémantika stavových formulí II

Vyhodnocování výrazů

Nechť s je stavem na V a e je výrazem nad V. Pak hodnota výrazu e ve stavu s *(značeno* $s[e]$*)* je definována indukcí.

- Hodnota proměnné $x \in V$ je $s[x]$.
- Pro výraz $f(e_1, \ldots, e_m)$, se definuje
$$s[f(e_1, \ldots, e_m)] = f(s[e_1], \ldots, s[e_m]).$$

Vyhodnocování booleovských formulí

Nechť s je stavem na V a φ je booleovskou formulí nad V. Pak hodnota booleovské formule φ ve stavu s *(značeno* $s[\varphi]$*)* se definuje:

- Pro atomickou formulí $P(e_1, \ldots, e_m)$
$$s[P(e_1, \ldots, e_m)] = P(s[e_1], \ldots, s[e_m]).$$
- Pro booleovskou formulí složenou pomocí logických spojek
$$s[\neg p] = \neg s[p] \qquad s[p * q] = s[p] * s[q],$$
kde $*$ je jednou ze spojek $\wedge, \vee, \rightarrow, \leftrightarrow$.

Sémantika stavových formulí III

Odlišnost

Nechť s, s' jsou stavy nad V a $x \in V$ je proměnná. Pak stavy s a s' jsou **x-odlišné** jestliže

$$s'[y] = s[y] \qquad \text{pro každé } y \in V - \{x\}.$$

Sémantika stavových formulí IV

Vyhodnocování stavových formulí (tvrzení)

Nechť s je stav a p, q jsou stavové formule, vše nad V. Pak říkáme, že formule p je splnitelné ve stavu s, značeno

$$s \models p,$$

a definujeme podle tvaru formule p pro:

- booleovskou formuli

  $s \models p$ právě když $s[p] = \mathrm{T}$,

- formule s kvantifikaci přes proměnnou u:
  - $s \models \exists u : p$ právě když $s' \models p$ pro nějaké s' nad V, které je u-odlišné od s,
  - $s \models \forall u : p$ právě když $s' \models p$ pro každé s' nad V, která jsou u-odlišné od s.
- formule spojené logickými spojkami:
  - $s \models \neg p$ právě když $s \not\models p$,
  - $s \models p \vee q$ právě když $s \models p$ nebo $s \models q$.

Nechť $s \models p$, pak říkáme, že s uspokojuje p a s označujeme jako p-stav.

Vlastností stavů

Splnitelnost

Libovolná stavová formule p je stavově splnitelná pokud existuje nějaké s takové, že $s \models p$.

Platnost

Stavová formule p se označuje za stavově platnou, pokud platí $s \models p$ pro každý stav s.

Ekvivalentnost

Nechť jsou p a q stavové formule. Pak říkáme, že p a q jsou stavově ekvivalentní, pokud platí $s \models p \wedge s \models q$ pro každý stav s.

Dostupnost

Nechť $\mathcal{C}$ je množina posloupností. Existuje-li posloupnost $\sigma \in \mathcal{C}, \sigma = s_0, s_1, \ldots, s_j, \ldots$ a pozice $j \geq 0$, taková, že $s = s_j$, pak nazýváme stav s $\mathcal{C}$-dostupným.

Temporální formule

Definice

Temporální formuli tvoří stavová formule, na kterou se aplikují temporální operátory, logické spojky a kvantifikátory.

Vyhodnocování

Nechť p je stavová formule a σ je modelem (posloupností stavů), obojí nad V. Pak říkáme, že formule p je splnitelná v čase t modelu σ, značeno

$$(\sigma, t) \models p,$$

pro $t \geq 0$ a definováno

$$(\sigma, t) \models p \qquad \text{právě když} \qquad s_t \Vdash p,$$

kde $\sigma : s_0, s_1, \ldots, s_t, \ldots$.

Temporální operátory – budoucí

Next $\bigcirc$

$(\sigma, t) \models \bigcirc p$ právě když $(\sigma, t+1) \models p$

t	0	1	2	3	4	5	6	7
x	0	0	1	1	0	1	0	1
$x = 0$	T	T	F	F	T	F	T	F
$\bigcirc(x = 0)$	T	F	F	T	F	T	F	...

Temporální operátory – budoucí

Henceforth $\Box$

$(\sigma, t) \models \Box p$ právě když $(\sigma, k) \models p$ **pro každé** $k \geq t$

t	0	1	2	3	4	5	6	7
x	4	7	3	6	5	3	4	3
$x \leq 5$	T	F	T	F	T	T	T	T
$\Box(x \leq 5)$	F	F	F	F	T	T	T	T

Temporální operátory – budoucí

Eventualy $\Diamond$

$(\sigma, t) \models \Diamond p$ právě když $(\sigma, k) \models p$ **pro nějaké** $k \geq t$

t	0	1	2	3	4	5	6	7
x	6	2	4	5	3	6	5	8
$x \geq 4$	F	T	T	T	F	F	F	F
$\Diamond(x \geq 4)$	F	F	F	F	T	F	T	F

Temporální operátory – budoucí

Until $\mathcal{U}$

$(\sigma, t) \models p\mathcal{U}q$ právě když **existuje** $k \geq t$ **takové, že** $(\sigma, k) \models q$,
a pro každé $i, j \leq i < k, (\sigma, i) \models p$

t	0	1	2	3	4	5	6	7
x	1	2	3	4	5	6	7	8
$2 \leq x \leq 4$	F	T	T	T	F	F	F	F
$x = 5$	F	F	F	F	T	F	T	F
$(2 \leq x \leq 4)\mathcal{U}(x = 5)$	F	T	T	T	T	F	F	F

Unless (Waiting-for) $\mathcal{W}$

$(\sigma, t) \models p\mathcal{W}q$ právě když $(\sigma, t) \models p\mathcal{U}q$ **or** $(\sigma, t) \models \Box p$

t	0	1	2	3	4	5	6	7
x	1	2	3	4	5	6	7	8
$(2 \leq x \leq 4) \vee (x > 6)$	F	T	T	T	F	F	T	T
$x = 5$	F	F	F	F	T	F	F	F
$[(2 \leq x \leq 4) \vee (x > 6)]\mathcal{W}(x = 5)$	F	T	T	T	T	F	T	T

Temporální operátory – minulé

Previous $\ominus$

$(\sigma, t) \models \ominus p$ právě když $(t > 0) \wedge (\sigma, t-1) \models p$

t	0	1	2	3	4	5	6	7
x	0	0	1	1	0	1	0	1
$x = 0$	T	T	F	F	T	F	T	F
$\ominus(x = 0)$	F	T	T	F	F	T	F	T

Temporální operátory – minulé

Has been always $\boxminus$

$(\sigma, t) \models \boxminus p$ právě když $(\sigma, t) \models p$ **pro každé** $k, 0 \leq k \leq j$

t	0	1	2	3	4	5	6	7
x	4	8	6	9	1	7	2	5
$(x \geq 4)$	T	T	T	T	F	T	F	T
$\boxminus(x \geq 4)$	T	T	T	T	F	F	F	F

Temporální operátory – minulé

Once

$(\sigma, t) \models \boxminus p$ právě když $(\sigma, t) \models p$ **pro nějaké** $k, 0 \leq k \leq j$

t	0	1	2	3	4	5	6	7
x	7	4	5	2	0	1	3	8
$(x < 3)$	F	F	F	T	T	T	F	F
$\diamondminus(x < 3)$	F	F	F	T	T	T	T	T

Temporální operátory – minulé

Since $\mathcal{S}$

$(\sigma, t) \models p\mathcal{S}q$ právě když **existuje** $i, 0 \leq i \leq t$ **takové, že** $(\sigma, i) \models q$ **a pro každé** $k, i \leq k \leq t, (\sigma, k) \models p$

t	0	1	2	3	4	5	6	7
x	1	2	3	4	5	6	7	8
$x \leq 5$	T	T	T	T	T	F	F	F
$x = 3$	F	F	T	F	F	F	F	F
$(x \leq 5)\mathcal{S}(x = 3)$	F	F	T	T	T	F	F	F

Back-to $\mathcal{B}$

$(\sigma, t) \models p\mathcal{B}q$ právě když $(\sigma, t) \models p\mathcal{S}q$ **or** $(\sigma, t) \models \boxminus p$

t	0	1	2	3	4	5	6	7
x	1	2	3	4	5	6	7	8
$x \neq 3$	T	T	F	T	T	T	T	T
$x = 6$	F	F	F	F	F	T	F	F
$(x \neq 3)\mathcal{B}(x = 6)$	T	T	F	F	F	T	T	T

Vlastnosti temporálních formulí

Odlišnost modelů

Nechť $\sigma : s_0, s_1, \ldots$ a $\sigma' : s'_0, s'_1, \ldots$ jsou modely nad V a $x \in V$ je proměnnou. Říkáme, že model σ' je x-**odlišný** od modelu σ jestliže pro každé $t \geq 0$ platí, že stav s'_t x-**odlišný** od stavu s_t.

Kvantifikace

Nechť p je formulí, u je proměnnou, σ a σ' jsou modely, vše nad V. Pak definujeme formuli kvantifikovanou

- existenčně

  $(\sigma, t) \models \exists u : p$ právě když $(\sigma, t) \models p$ pro nějaký model σ', který je u-**odlišný** od modelu σ

- univerzálně

  $(\sigma, t) \models \forall u : p$ právě když $(\sigma, t) \models p$ pro každý model σ', který je u-**odlišný** od modelu σ

Vlastnosti temporálních formulí

Uspokojitelnost

Nechť platí $(\sigma, t) \models p$, pak říkáme, že model σ uspokojuje p na pozici t a t označujeme jako p-**pozici**.

Je-li formule p splnitelná na pozici 0 modelu σ, pak zapisujeme $\sigma \models p$ a říkáme, že model σ uspokojuje formuli p.

Splnitelnost

Formule p nad V nazýváme splnitelnou, jestliže platí $\sigma \models p$ pro nějaký model σ.

Platnost

Formule p nad V nazýváme platnou, jestliže platí $\sigma \models p$ pro každý model σ.

Vlastnosti temporálních formulí

Ekvivalence

Nechť p a q jsou dvě formule nad V. Říkáme, že p a q jsou ekvivalentní, značíme $p \sim q$, právě když $p \leftrightarrow q$.

Kongruence

Nechť p a q jsou dvě formule nad V. Říkáme, že p a q jsou kongruentní, značíme $p \approx q$, právě když $\Box(p \leftrightarrow q)$.

Vlastnosti temporálních formulí

Substituce

Nechť $\varphi(u)$ je formuli s nějakými výskyty symbolu u a p je formulí, vše nad V. Pak substitucí p za u, značeno $\varphi[p/u]$, rozumíme nahrazení všech výskytu symbolu u ve formuli φ formuli p.

Stavová substituce

Nechť $\varphi(u)$ je stavovou formuli s nějakými výskyty symbolu u a p, q jsou obecně formulemi, vše nad V. A platí-li $p \sim q$, pak zároveň platí $\varphi(p) \sim \varphi(q)$.

Temporální substituce

Nechť $\varphi(u)$ je temporální formuli s nějakými výskyty symbolu u a p, q jsou obecně formulemi, vše nad V. A platí-li $p \approx q$, pak zároveň platí $\varphi(p) \approx \varphi(q)$.

Slabší verze operátoru Previous $\widetilde{\ominus}$

$$(\sigma, t) \models \widetilde{\ominus} p \quad \text{právě když} \quad t = 0 \vee (j > 0 \wedge (\sigma, t-1) \models p)$$

Základní množina operátorů

Slabší verze operátoru Previous $\widetilde{\ominus}$

$$(\sigma, t) \models \widetilde{\ominus} p \quad \text{právě když} \quad t = 0 \vee (j > 0 \wedge (\sigma, t-1) \models p)$$

Základní množina

$$\neg, \vee, \bigcirc, \mathcal{W}, \widetilde{\ominus}, \mathcal{B}$$

Základní množina operátorů

Slabší verze operátoru Previous $\widetilde{\ominus}$

$$(\sigma, t) \models \widetilde{\ominus} p \quad \text{právě když} \quad t = 0 \vee (j > 0 \wedge (\sigma, t-1) \models p)$$

Základní množina

$$\neg, \vee, \bigcirc, \mathcal{W}, \widetilde{\ominus}, \mathcal{B}$$

Vztahy

$$\Box p \approx p \mathcal{W} \mathbf{F}$$
$$\Diamond p \approx \neg \Box \neg p$$
$$p \mathcal{U} q \approx (p \mathcal{W} q) \wedge \Diamond q$$

$$\boxminus p \approx p \mathcal{B} \mathbf{F}$$
$$\diamondminus p \approx \neg \boxminus \neg p$$
$$p \mathcal{S} q \approx (p \mathcal{B} q) \wedge \diamondminus q$$
$$\ominus p \approx \neg \widetilde{\ominus} \neg p$$

Vlastnosti temporálních operátorů

Dualita

- Budoucí operátory

$$\neg \Box p \approx \Diamond \neg p \qquad \neg \Diamond p \approx \Box \neg p$$

$$\neg (p \mathcal{U} q) \approx (\neg q) \mathcal{W} (\neg p \wedge q) \qquad \neg (p \mathcal{W} q) \approx (\neg q) \mathcal{U} (\neg q \wedge \neg q)$$

$$\neg \bigcirc p \approx \bigcirc \neg p$$

- Minulé operátory

$$\neg \boxminus p \approx \diamondminus \neg p \qquad \neg \diamondminus p \approx \boxminus \neg p$$

$$\neg (p \mathcal{S} q) \approx (\neg q) \mathcal{B} (\neg p \wedge q) \qquad \neg (p \mathcal{B} q) \approx (\neg q) \mathcal{S} (\neg q \wedge \neg q)$$

$$\neg \ominus p \approx \widetilde{\ominus} \neg p \qquad \neg \widetilde{\ominus} p \approx \widetilde{\ominus} \neg p$$

Vlastnosti temporálních operátorů

Idempotence

- Budoucí operátory

$$\Box\Box p \approx \Box p$$
$$\Diamond\Diamond p \approx \Diamond p$$
$$p\mathcal{U}(p\mathcal{U}q) \approx p\mathcal{U}q \qquad p\mathcal{W}(p\mathcal{W}q) \approx p\mathcal{W}q$$
$$(p\mathcal{U}q)\mathcal{U}q \approx p\mathcal{U}q \qquad (p\mathcal{W}q)\mathcal{W}q \approx p\mathcal{W}q$$

- Minulé operátory

$$\boxminus\boxminus p \approx \boxminus p$$
$$\diamondminus\diamondminus p \approx \diamondminus p$$
$$p\mathcal{S}(p\mathcal{S}q) \approx p\mathcal{S}q \qquad p\mathcal{B}(p\mathcal{B}q) \approx p\mathcal{B}q$$
$$(p\mathcal{S}q)\mathcal{S}q \approx p\mathcal{S}q \qquad (p\mathcal{B}q)\mathcal{B}q \approx p\mathcal{B}q$$

Vlastnosti temporálních operátorů

Absorpce

- Budoucí operátory

$$\Diamond\Box\Diamond p \approx \Box\Diamond p$$

$$\Box\Diamond\Box p \approx \Diamond\Box p$$

$$p\mathcal{W}(p\mathcal{U}q) \approx p\mathcal{W}q \qquad (p\mathcal{U}q)\mathcal{W}q \approx p\mathcal{U}q$$

$$p\mathcal{U}(q\mathcal{W}q) \approx p\mathcal{W}q \qquad (p\mathcal{W}q)\mathcal{U}q \approx p\mathcal{U}q$$

- Minulé operátory

$$\diamondminus\boxminus\diamondminus p \approx \boxminus\diamondminus p$$

$$\boxminus\diamondminus\boxminus p \approx \diamondminus\boxminus p$$

$$p\mathcal{B}(p\mathcal{S}q) \approx p\mathcal{B}q \qquad (p\mathcal{S}p)\mathcal{B}q \approx p\mathcal{S}q$$

$$p\mathcal{S}(q\mathcal{B}q) \approx p\mathcal{B}q \qquad (p\mathcal{B}q)\mathcal{S}q \approx p\mathcal{S}q$$

Vlastnosti temporálních operátorů

Komutativní vlastnosti operátoru Next

Nechť $\varphi_{NP}(p_1, \ldots, p_n)$ je formulí, která neobsahuje žádný minulý temporální operátor, pak platí:

$$\bigcirc\varphi_{NP}(p_1, \ldots, \_n) \Leftrightarrow \varphi_{NP}(\bigcirc p_1, \ldots, \bigcirc p_n).$$

Vlastnosti temporálních operátorů

Pozitivní/Negativní výskyty

Vyskytuje-li se podformule p ve formuli φ, a je-li počet negovaných výskytu podformule p

sudý , pak říkáme, že má pozitivní výskyt,

lichý , pak říkáme, že má negativní výskyt

ve formuli φ.

Komutativní vlastnosti operátoru Previous

Nechť $\varphi_{NF}(p_1, \ldots, p_m, q_1, \ldots, q_n)$ je formulí, která neobsahuje žádný budoucí temporální operátor, formule $p_1, \ldots, p_m$ jsou všechny pozitivní výskyty podformulí a $q_1, \ldots, q_n$ jsou všechny negativní výskyty podformulí, pak platí:

$$\ominus\, \varphi_{NF}(p_1, \ldots, \_m, q_1, \ldots, q_n) \Leftrightarrow \varphi_{NF}(\ominus\, p_1, \ldots, \ominus\, p_m, \widetilde{\ominus} q_1, \ldots, \widetilde{\ominus} q_n),$$
$$\widetilde{\ominus} \varphi_{NF}(p_1, \ldots, \_m, q_1, \ldots, q_n) \Leftrightarrow \varphi_{NF}(\widetilde{\ominus} p_1, \ldots, \widetilde{\ominus} p_m, \ominus\, q_1, \ldots, \ominus\, q_n).$$

Axiomy

Budoucí axiomy

FX0	$\Box \rightarrow p$
FX1	$\bigcirc\neg p \Leftrightarrow \neg \bigcirc p$
FX2	$\bigcirc(p \rightarrow q) \Leftrightarrow (\bigcirc p \rightarrow \bigcirc q)$
FX3	$\Box(p \rightarrow q) \Leftrightarrow (\Box p \rightarrow \Box q)$
FX4	$\Box p \rightarrow \Box \bigcirc p$
FX5	$(p \Rightarrow \bigcirc p) \rightarrow (p \Rightarrow \Box p)$
FX6	$p \mathcal{W} q \Leftrightarrow [q \vee (p \wedge \bigcirc(p \mathcal{W} q))]$
FX7	$\Box p \Rightarrow p \mathcal{W} q$

Axiomy

Budoucí axiomy

PX0	$\ominus p \Rightarrow \widetilde{\ominus} p$
PX1	$\widetilde{\ominus}(p \rightarrow q) \Leftrightarrow (\widetilde{\ominus} p \rightarrow \widetilde{\ominus} q)$
PX2	$\boxminus(p \rightarrow q) \Rightarrow (\boxminus p \rightarrow \boxminus q)$
PX3	$\Box p \rightarrow \Box \widetilde{\ominus} p$
PX4	$(p \Rightarrow \widetilde{\ominus} p) \rightarrow (p \Rightarrow \boxminus p)$
PX5	$p \mathcal{B} q \Leftrightarrow (q \vee [p \wedge \widetilde{\ominus}(p \mathcal{B} q)])$
PX6	$\widetilde{\ominus} \mathbf{F}$

Kombinované axiomy

FX8	$\bigcirc \ominus p$
PX7	$\widetilde{\ominus} \bigcirc p$

Odvozovací pravidla

Definice

Nechť $p_1, \ldots, p_n$ a q jsou formulemi nad V. Existuje-li způsob, jak z formulí $p_1, \ldots, p_n$ odvodit formuli q, značeno

$$\frac{p_1, \ldots p_n}{q} \text{ nebo } p_1, \ldots, p_n \vdash q,$$

pak formule $p_1, \ldots, p_n$ nazýváme předpoklady a formuli q úsudkem.

Odvozovací pravidla

Pravidlo zobecnění — GEN

$$\frac{\Vdash p}{\models \Box p}$$

Pravidlo specializace — SPEC

$$\frac{\models \Box p}{\Vdash p}$$

Pravidlo konkretizace — INST

$$\frac{p}{p[q/u]}$$

Pravidlo Modus Ponens — MP

$$\frac{(p_1 \wedge \cdots \wedge p_n) \rightarrow q, p_1, \ldots, p_n}{q}$$

- **Úvod do temporální logiky**
  - Stavová formule
  - Temporální formule
  - Dokazovací systém
- **Vlastnosti programů**
  - Přechodový systém
  - Lokální jazyk
  - Klasifikace vlastností
- **Příklad**

Přechodový systém

Přechodový systém

Přechodový systém je čtveřice

$$P = (\Pi, \Sigma, \mathcal{T}, \Theta),$$

kde

- Π je konečná množina stavových proměnných,
- Σ je množina stavů,
- $\mathcal{T}$ je konečnou množinou přechodů ve tvaru $\tau : \Sigma \to 2^{\Sigma}, \tau \in \mathcal{T}$, aplikací přechodu τ na stav $s \in \Sigma$ získáme množinu stavů (může být prázdná), do kterých lze přejít. Každý stav s', do kterého lze přejít, nazýváme τ-následníkem.
- Θ je startovací podmínkou.

Přechodový systém

Nečinný přechod

Nechť $\tau \in \mathcal{T}$ je přechodem, a $s \in \Sigma$ je stavem přechodového systému, pokud platí,

$$\tau(s) = \{s\},$$

pak takovýto přechod nazýváme nečinným a označujeme τ_I.

Přechodový systém

Relace přechodu

Nechť $\tau \in \mathcal{T}$ je přechodem, pak $\rho_\tau(\Pi, \Pi')$ nazýváme relaci přechodu:

$$\rho_\tau(\Pi, \Pi') : C_\tau(\Pi) \wedge (y'_1 = e_1) \wedge \cdots \wedge (y'_k = e_k),$$

kde

$C_\tau(\Pi)$ je povolující podmínkou, tzn. ohodnocení proměnných z množiny Π umožňuje provedeni přechodu τ,

$(y'_1 = e_1) \wedge \cdots \wedge (y'_k = e_k)$ jsou modifikační příkazy, pro které platí

- $y_n \in \Pi$, pro $n = 1 \ldots k$ a
- e_n je nová hodnota.

Lokální jazyk

Definice

Lokální jazyk je jazykem prvního řádu nad dvěma typy predikátu: stavovými a přechodovými.

Predikáty

Stavové – vyhodnocují se nad stavem.

Přechodové – vyhodnocují se nad dvojici stav a jeho přímý předchůdce.

Program

Dále uvažujme, že π je množina všech příkazu, resp. návěští, popisovaného programu.

Lokální jazyk

Lokační predikáty

Mějme zápis příkazu ve formě $\ell : S : \tilde{\ell}$, kde ℓ, resp. $\tilde{\ell}$, jsou návěští ohraničující příkaz S zepředu, resp. zezadu. Pak můžeme pozici v programu označit pomocí predikátu:

- at_ℓ, at_S — abychom vyjádřili, že se nacházíme ve stavu před provedením tohoto příkazu,
- at_$\tilde{\ell}$, after_S — pro vyjádřeni stavu, ve kterém již byl tento program vykonán,
- in_S — v případě, že příkaz S označuje blok příkazu a program se nachází uvnitř tohoto bloku.

Enabled

Definujme predikát

$$\text{enabled}(\tau)\colon C_\tau$$

který vyjadřuje, zda je možné daný přechod realizovat a $T \subseteq \mathcal{T}$ jako množinu všech přechodů, které lze v daném stavu realizovat.

$$\text{enabled}(T)\colon \bigvee_{\tau \in T} \text{enabled}(\tau).$$

Terminal

Mějme predikát terminal, vyjadřující stav, když již nelze provést další přechod

$$\text{terminal} = \bigwedge_{\tau \in \mathcal{T} - \{\tau_I\}} \neg\text{enabled}(\tau).$$

Přechodové predikáty

Mějme přechodovou formuli

$$\neg\text{first} \wedge \varphi(\Pi^-, \Pi),$$

kde φ je relaci libovolného přechodu $\tau \in \mathcal{T}$ a notace Π^- označuje ohodnoceni proměnných v přímo předcházejícím stavu.

Pak definujeme přechodový predikát last-taken následovně

$$\text{last-taken: } \neg\text{first} \wedge \rho_\tau(\Pi^-, \Pi).$$

Komunikační predikáty – Asynchronní

- Zaslání zprávy ν do kanálu α

$$[\alpha \leftarrow \nu] : \neg\text{first} \wedge (\alpha = \alpha^- \bullet \nu)$$

- Přečtení zprávy ν z kanálu α

$$[\alpha \rightarrow \nu] : \neg\text{first} \wedge (\nu \bullet \alpha = \alpha^-)$$

Komunikační predikáty – Synchronní

Nechť $\tau_{\langle \ell, m \rangle}$ je komunikačním přechodem spojený s provedením dvou příkazů:

$$\ell : \alpha \leftarrow e \qquad m : \alpha \rightarrow \nu.$$

Pak definujeme

$$\text{comm}(\ell, m, \nu) : \text{last-taken}(\tau_{\langle \ell, m \rangle}) \wedge (\nu = e^-).$$

$$[\alpha \leftarrow \nu] : \bigvee_{\langle \ell, m \rangle} \text{comm}(\ell, m, \nu)$$

Klasifikace vlastností

Třídy vlastností

Vlastnosti dělíme do 6 tříd, a každá z těchto vlastností je charakterizována kanonickou temporální formulí.

- Bezpečnost
- Zaručenost
- Obligátnost
- Odpovědnost
- Stabilita
- Reaktivnost

Klasifikace vlastností

Bezpečné vs. Průběhové vlastnosti

Každá třída, jejíž kanonická formule obsahuje operátor $\Diamond$ poskytuje tzv. průběhové vlastností.

Bezpečné vs. Živé vlastnosti

Neformálně lze toto dělení popsat:

bezpečné vlastnosti garantují, že se nestane něco špatného

živé vlastnosti garantují, že nastane něco dobrého

Třídy vlastností

Nechť Σ je množinou všech stavů programu a Σ^ω je potenční množinou všech posloupností stavů. Vlastnost $\mathcal{P}$ je nějakou podmnožinou Σ^ω, pro kterou platí, že každá posloupnost z množiny $\mathcal{P}$ uspokojuje formuli p, která definuje vlastnost třídy.

$$\mathcal{P} = \{\sigma | \sigma \in \Sigma^\omega, \sigma \models p\}$$

Mějme třídy $\mathcal{P}$ danou vlastností p a třídu $\mathcal{Q}$ danou vlastnosti q. Pak dostáváme aplikaci logických spojek na definující formule operace nad množinami tříd:

$p \wedge q$	definuje	$\mathcal{P} \cap \mathcal{Q}$
$p \vee q$	definuje	$\mathcal{P} \cup \mathcal{Q}$
$\neg p$	definuje	$\overline{\mathcal{P}} = \Sigma^\omega - \mathcal{P}$

Dále zkoumejme uzávěrové vlastnosti uvedených tříd.

Bezpečnost

$$\Box p,$$

pro minulou formuli p.

Uzávěrové vlastnosti

Třída bezpečných vlastností je uzavřená vůči pozitivním množinovým operacím, tedy průniku a sjednocení.

$$(\Box p \wedge \Box q) \quad \sim \quad \Box(p \wedge q)$$
$$(\Box p \vee \Box q) \quad \sim \quad \Box(\boxminus \vee \boxminus q)$$

Příklady vlastností

- Komplexní invariance
- Částečná správnost
- Neuváznutí, příp. místní neuváznutí
- Bezchybovost
- Vzájemné vyloučení

Zaručenost

$$\Diamond p,$$

pro minulou formuli p.

Uzávěrové vlastnosti

Třída zaručujících vlastností je uzavřená vůči pozitivním množinovým operacím, tedy průniku a sjednocení.

$$\begin{array}{lcl} (\Diamond p \vee \Diamond q) & \sim & \Diamond(p \vee q) \\ (\Diamond p \wedge \Diamond q) & \sim & \Diamond(\diamondminus \wedge \diamondminus q) \end{array}$$

Obligátnost

$$\Box p \vee \Diamond q,$$

pro minulou formuli p. V tomto tvaru nazýváme prostou obligátní formuli. Kanonická formule má tvar

$$\bigwedge_{i=1}^{n} (\Box p_i \vee \Diamond q_i),$$

kde $p_i, q_i, i = 1, \ldots, n$ jsou minulé formule.

Uzávěrové vlastnosti

Je zřejmé, že každá obligátní formule je kombinací bezpečné a zaručené formule a tedy platí stejné uzávěrové vlastnosti.

Odpovědnost

$$\Box \Diamond p,$$

pro minulou formuli p.

Uzávěrové vlastnosti

Třída odpovědných vlastností je uzavřená vůči pozitivním množinovým operacím, tedy průniku a sjednocení.

$$\begin{array}{lcl} (\Diamond p \vee \Diamond q) & \sim & \Diamond(p \vee q) \\ (\Diamond p \wedge \Diamond q) & \sim & \Diamond(\diamondminus p \wedge \diamondminus q) \end{array}$$

Stabilita

$$\Diamond \Box p,$$

pro minulou formuli p.

Uzávěrové vlastnosti

Třída stabilních vlastností je uzavřená vůči pozitivním množinovým operacím, tedy průniku a sjednocení.

$$(\Diamond \Box p \wedge \Diamond \Box q) \quad \sim \quad \Diamond \Box (p \wedge q)$$
$$(\Diamond \Box p \vee \Diamond \Box q) \quad \sim \quad \Diamond \Box \big(q \wedge \ominus(p \mathcal{S} (p \wedge (\neg q)))\big)$$

Reaktivnost

$$\Box \Diamond p \vee \Diamond \Box q,$$

pro minulou formuli p. V tomto tvaru nazýváme kanonickou prostou reaktivní formuli. Obecná reaktivní formule má tvar

$$\bigwedge_{i=1}^{n} (\Box \Diamond p_i \vee \Diamond \Box q_i),$$

kde $p_i, q_i, i = 1, \ldots, n$ jsou minulé formule.

Uzávěrové vlastnosti

Je zřejmé, že každá reaktivní formule je kombinací odpovědné a stabilní formule a tedy platí stejné uzávěrové vlastnosti.

- **Úvod do temporální logiky**
  - Stavová formule
  - Temporální formule
  - Dokazovací systém
- **Vlastnosti programů**
  - Přechodový systém
  - Lokální jazyk
  - Klasifikace vlastností
- **Příklad**

Příklad I

FIFO buffer – bez duplicitních zpráv

Seznam vlastností, které by měl buffer splňovat.

- Každá zpráva přijatá na vstupu α, by musí být někdy odeslána na výstup β
  $[\alpha \rightarrow m] \Rightarrow \Diamond[\beta \leftarrow m]$
- Každá zpráva odeslána na vstup β, musela být nejprve přijata na vstupu α
  $[\beta \leftarrow m] \Rightarrow \diamondminus[\alpha \rightarrow m]$
- Každá přijatá zpráva je vyslána na výstup nejvýše jednou
  $[\beta \leftarrow m] \Rightarrow \ominus \boxminus \neg[\beta \leftarrow m]$
- Zachování stejného pořadí zpráv ze vstupu na výstupu
  $\big((\neg[\beta \leftarrow m'])\mathcal{U}[\beta \leftarrow m]\big) \rightarrow \big((\neg[\alpha \leftarrow m']\mathcal{W}[\alpha \leftarrow m]\big)$

Reference

- Z. Manna, A. Pnueli: The Temporal Logic of Reactive and Concurrent Systems, Springer-Verlag. 1992. New York. ISBN 0-387-97664-7

Děkuji za pozornost!

Konec