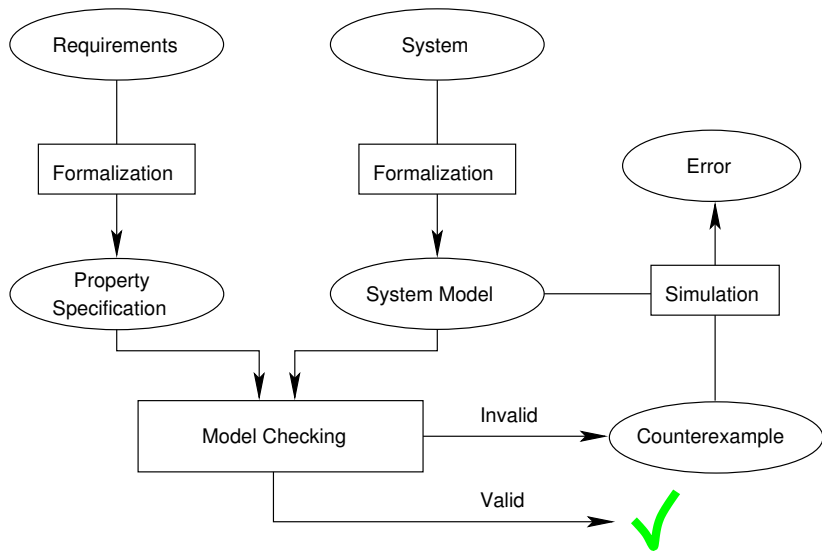


IV113 Validace a verifikace

Ověřování modelu pro LTL

doc. RNDr. Jiří Barnat, Ph.D.

Schéma metody



Formalizace systému

Připomenutí

- Na systém lze nahlížet jako na množinu stavů, které se mění vlivem vykonávání akcí programu.
- Stav = valuace modelovaných proměnných.

Atomické propozice

- Základní tvrzení vyjadřujících se o kvalitách jednotlivých stavů. (Např. $\max(x, y) \geq 3, \dots$)
- Platnost atomických propozic musí být algoritmicky rozhodnutelná na základě daného stavu, tj. s využitím valuace modelovaných proměnných.
- Množina základních o stavu pozorovatelných jevů je ovlivněna mírou abstrakce použitou při modelování systému.

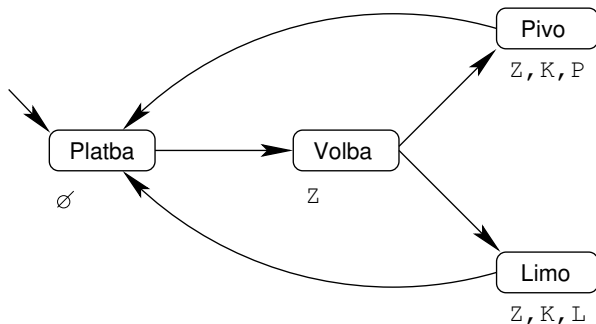
Kripkeho struktury

- Nechť je dána množina atomických propozic AP .
- Kripkeho struktura je čtveřice (S, T, I, s_0) , kde
 - S je (konečná) množina stavů,
 - $T \subseteq S \times S$ je přechodová relace,
 - $I : S \rightarrow 2^{AP}$ je interpretace AP .
 - $s_0 \in S$ je počáteční stav.

Kripkeho přechodové systémy

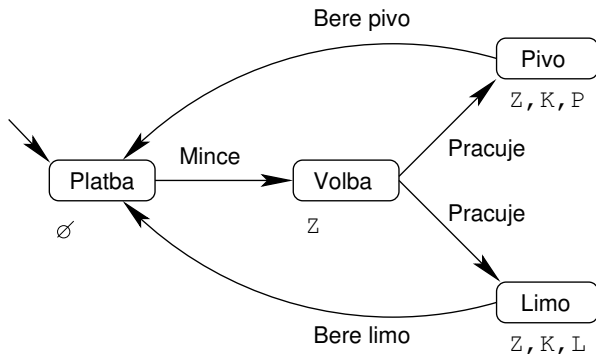
- Je-li dána množina Act akcí proveditelných programem, je možné Kripkeho struktury rozšířit o označení přechodů.
- Kripkeho přechodový systém je pětice $(S, T, I, s_0, \mathcal{L})$, kde
 - (S, T, I, s_0) je Kripkeho struktura,
 - $\mathcal{L} : T \rightarrow Act$ je značkovácí funkce.

Kripkeho struktura



$AP = \{Z - \text{zaplaceno}, K - \text{kelímek}, L - \text{limo}, P - \text{pivo}\}$

Kripkeho přechodový systém



$AP = \{Z - \text{zaplaceno}, K - \text{kelímek}, L - \text{limo}, P - \text{pivo}\}$

Běh

- Maximální (tj. nerozšiřitelná) cesta v grafu indukovaného Kripkeho strukturou, počínající v iniciálním stavu Kripkeho struktury.
- Nechť $M = (S, T, I, s_0)$ je Kripkeho struktura.
- Běh je posloupnost stavů $\pi = s_0, s_1, s_2, \dots$ taková, že $\forall i \in \mathbb{N}_0. (s_i, s_{i+1}) \in T$.

Konečné běhy – úmluva

- Maximální cesta může být konečná: $\pi = s_0, s_1, s_2, \dots, s_k$, pokud $\nexists s_{k+1} \in S. (s_k, s_{k+1}) \in T$.
- Z technických důvodů lze na konečné maximální cesty nahlížet jako na nekonečné běhy, které vzniknou opakováním posledního stavu dané maximální cesty.
- Maximální cesta s_0, \dots, s_k se chápe jako běh $s_0, \dots, s_k, s_k, s_k, \dots$

Formalizace vlastností

Problém

- Jak se formálně vyjadřovat o kvalitách běhu?
- Jak mechanicky rozhodovat, že běh má danou kvalitu?

Řešení

- Konečný automat jako mechanický pozorovatel běhu.
- Běh je nekonečný!
- Konečné automaty pro jazyky nekonečných slov (ω -regulární jazyky)
- Büchi akceptační podmínka – automat akceptuje slovo pokud nekonečněkrát projde koncovým stavem.

Büchi automaty

- Büchi automat je pětice $A = (\Sigma, S, s, \delta, F)$, kde
 - Σ je konečná abeceda znaků,
 - S je konečná množina stavů,
 - $s \in S$ je iniciální stav,
 - $\delta : S \times \Sigma \rightarrow 2^S$ je přechodová relace, a
 - $F \subseteq S$ je množina koncových stavů.

Jazyk akceptovaný Büchi automatem A

- Běh ρ automatu A nad nekonečným slovem $w = a_1 a_2 \dots$ je sekvence stavů $\rho = s_0, s_1, \dots$ taková, že $s_0 \equiv s$ a $\forall i : s_i \in \delta(s_{i-1}, a_i)$.
- $\text{inf}(\rho)$ – množinu stavů, které se v ρ vyskytly nekonečně krát.
- Běh ρ je akceptující, pokud $\text{inf}(\rho) \cap F \neq \emptyset$.
- Jazyk akceptovaný automatem A je množina všech slov, pro které existuje akceptující běh. Označujeme $L(A)$.

Pozorování

- $AP = \{X, Y, Z\}$,
- Hrana označená $\{X\}$ značí, že platí X a neplatí Y a Z .
- Pokud je třeba vyjádřit že platí X , neplatí Z a na platnosti Y nezáleží, je třeba vést hrany pod $\{X\}$ a $\{X, Y\}$.

Množiny AP jako boolovské formule

- Jednotlivé hrany mezi dvěma stejnými vrcholy pod různými kombinacemi atomických propozic lze sloučit do jedné hrany ohodnocené boolovskou formulí.

Příklad

- Hrany $\{X\}$, $\{Y\}$, $\{X, Y\}$, $\{X, Z\}$, $\{Y, Z\}$ a $\{X, Y, Z\}$ lze nahradit jednou hranou ohodnocenou formulí $X \vee Y$.
- Pokud vůbec nezáleží na platnosti při provedení hrany, je možné ji označit štítkem $true \equiv X \vee \neg X$.

System

- Prodejní automat
- $\Sigma = 2^{\{Z,K,L,P\}}$,
- $jeZ = \{A \in \Sigma \mid Z \in A\}$, $jeK = \{A \in \Sigma \mid K \in A\}$, ...

Vlastnosti

- Prodejní automat vydá alespoň jeden kelímek s nápojem.
- Prodejní automat vydá alespoň jeden kelímek s limonádou.
- Prodejní automat vydá nekonečně mnoho kelímků.
- Prodejní automat vydá nekonečně mnoho kelímků s pivem.
- Prodejní automat nevydá kelímek s nápojem, bez zaplacení.
- Pokaždé, když zaplatím, dostanu kelímek s nápojem.

Lineární temporální logika

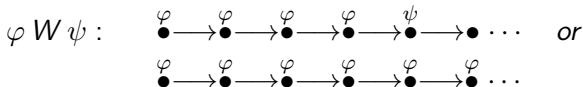
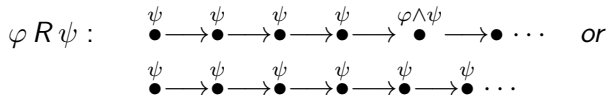
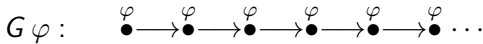
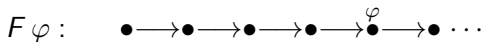
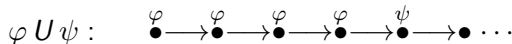
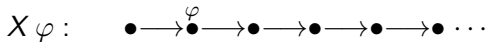
Formule φ

- Vyhodnocuje se nad jedním během systému.
- Vyjadřuje se o platnosti atomických propozic ve stavech daného běhu.

Temporální operátory LTL

- $F\varphi$ — (Future) Někde v běhu platí φ .
- $G\varphi$ — (Globally) V daném běhu vždy platí φ .
- $\varphi U \psi$ — (Until) Někde platí ψ a do té doby platí φ .
- $X\varphi$ — (Next) V příštím stavu platí φ .
- $\varphi W \psi$ — (Weak Until) Jako Until ale ψ nemusí nastat.
- $\varphi R \psi$ — (Release) ψ platí dokud neplatí $\varphi \wedge \psi$.

Grafické znázornění sémantiky temporálních operátorů



Nechť AP je množina atomických propozic. Pak

- Je-li $p \in AP$, pak p je formule.
- Je-li φ formule, pak $\neg\varphi$ je formule.
- Jsou-li φ a ψ formule, pak $\varphi \vee \psi$ je formule.
- Je-li φ formule, pak $X\varphi$ je formule.
- Jsou-li φ a ψ formule, pak $\varphi U\psi$ je formule.

Alternativní zápis

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid X\varphi \mid \varphi U\varphi$$

Výroková logika

- $\varphi \wedge \psi \equiv \neg(\neg\varphi \vee \neg\psi)$
- $\varphi \Rightarrow \psi \equiv \neg\varphi \vee \psi$
- $\varphi \Leftrightarrow \psi \equiv (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$

Temporální operátory

- $F\varphi \equiv \text{true } U \varphi$
- $G\varphi \equiv \neg F\neg\varphi$
- $\varphi R \psi \equiv \neg(\neg\varphi U \neg\psi)$
- $\varphi W \psi \equiv \varphi U \psi \vee G\varphi$

Alternativní syntax

- $F\varphi \equiv \diamond\varphi$
- $G\varphi \equiv \square\varphi$
- $X\varphi \equiv \circ\varphi$

Model LTL formule

- Je dána množina atomických propozic AP .
- Modelem LTL formule je běh π Kripkeho struktury.

Značení

- Nechť $\pi = s_0, s_1, s_2, \dots$
- Suffix běhu π počínaje stavem s_k budeme značit jako $\pi^k = s_k, s_{k+1}, s_{k+2}, \dots$
- k -tý stav běhu π , budeme značit jako $\pi(k) = s_k$.

Předpoklady

- Je dána množina atomických propozic AP .
- Je dán běh π Kripkeho struktury $M = (S, T, I, s_0)$.
- φ, ψ jsou syntakticky správné LTL formule.
- $p \in AP$ je atomická propozice.

Sémantika

$$\begin{aligned}\pi \models p & \text{ iff } p \in I(\pi(0)) \\ \pi \models \neg\varphi & \text{ iff } \pi \not\models \varphi \\ \pi \models \varphi \vee \psi & \text{ iff } \pi \models \varphi \text{ or } \pi \models \psi \\ \pi \models X\varphi & \text{ iff } \pi^1 \models \varphi \\ \pi \models \varphi U\psi & \text{ iff } \exists k. 0 \leq k, \pi^k \models \psi \text{ and} \\ & \forall i. 0 \leq i < k, \pi^i \models \varphi\end{aligned}$$

$$\pi \models F \varphi \quad \text{iff} \quad \exists k. k \geq 0, \pi^k \models \varphi$$

$$\pi \models G \varphi \quad \text{iff} \quad \forall k. k \geq 0, \pi^k \models \varphi$$

$$\begin{aligned} \pi \models \varphi R \psi \quad \text{iff} \quad & (\exists k. 0 \leq k, \pi^k \models \varphi \wedge \psi \text{ and} \\ & \forall i. 0 \leq i < k, \pi^i \models \psi) \\ & \text{or } (\forall k. k \geq 0, \pi^k \models \psi) \end{aligned}$$

$$\begin{aligned} \pi \models \varphi W \psi \quad \text{iff} \quad & (\exists k. 0 \leq k, \pi^k \models \psi \text{ and} \\ & \forall i. 0 \leq i < k, \pi^i \models \varphi) \\ & \text{or } (\forall k. k \geq 0, \pi^k \models \varphi) \end{aligned}$$

Verifikace s použitím LTL

- Na systém nahlížíme jako na množinu možných běhů.
- Systém splňuje vlastnost specifikovanou LTL formulí pokud (a jen tehdy) všechny běhy systému začínající v iniciálním stavu systému splňují danou formuli.
- Kterýkoliv běh systému, který začíná v iniciálním stavu a nespĺňuje danou formuli může být dokladem toho, že systém nespĺňuje specifikovanou vlastnost.

Tvrzení

- Pokud konečně stavový systém nespĺňuje vlastnost specifikovanou formulí LTL, tak to lze dokladovat během π , který lze vyjádřit jako $\pi = \pi_1 \cdot (\pi_2)^\omega$, kde

$$\pi_1 = s_0, s_1, \dots, s_k$$

$$\pi_2 = s_{k+1}, s_{k+2}, \dots, s_{k+n}, \text{ kde } s_k \equiv s_{k+n}.$$

- Běhy s uvedenou vlastností nazýváme *lasa* (lasso shape).

LTL Model Checking s využitím teorie formálních jazyků a automatů

Pozorování 1

- Systém je množina (nekonečných) běhů.
- Na systém lze nahlížet jako na jazyk nekonečných slov.

Pozorování 2

- Dva různé běhy (různé posloupnosti stavů) jsou z hlediska platnosti dané formule ekvivalentní, pokud se shodují v interpretaci atomických proměnných.
- Jestliže $\pi = s_0, s_1, \dots$, pak $I(\pi) \stackrel{def}{\iff} I(s_0), I(s_1), I(s_2), \dots$

Pozorování 3

- Každý běh danou formuli buď splňuje, anebo nesplňuje.
- Každá LTL formule vymezuje množinu splňujících běhů.

Problém

- Je dána množina AP , Kripkeho struktura $M = (S, T, I, s_0)$ a specifikace LTL formulí φ .
- Splňuje systém M specifikaci φ ? ($M \models \varphi$)

Jazyky nekonečných slov

- Nechť $\Sigma = 2^{AP}$.
- Jazyk L_{sys} všech běhů systému M :

$$L_{\text{sys}} = \{I(\pi) \mid \pi \text{ je běh v } M\}.$$

- Jazyk L_{φ} všech běhů splňujících formuli φ :

$$L_{\varphi} = \{I(\pi) \mid \pi \models \varphi\}.$$

Pozorování

- Systém M splňuje specifikaci φ právě když $L_{\text{sys}} \subseteq L_{\varphi}$.

Tvrzení 1

- Pro každou LTL formuli φ existuje (a lze efektivně sestrojít) Büchi automat A_φ takový, že $L_\varphi = L(A_\varphi)$.
- [Vardi, Wolper 1986]

Tvrzení 2

- Pro každou Kripkeho strukturu $M = (S, T, I, s_0)$ lze sestrojít Büchi automat A_{sys} takový, že $L_{sys} = L(A_{sys})$.
- Konstrukce A_{sys}
 - Nechť AP je množina atomických propozic.
 - Pak $A_{sys} = (S, 2^{AP}, s_0, \delta, S)$, kde
 - $q \in \delta(p, a)$ právě když $(p, q) \in T \wedge I(p) = a$.

Tvrzení

Nechť $A = (S_A, \Sigma, s_A, \delta_A, F_A)$ a $B = (S_B, \Sigma, s_B, \delta_B, F_B)$ jsou Büchi automaty nad shodnou abecedou Σ . Pak existuje (lze efektivně sestrojít) Büchi automat $A \times B$ takový, že $L(A \times B) = L(A) \cap L(B)$.

- Konstrukce $A \times B$

- $A \times B = (S_A \times S_B \times \{0, 1\}, \Sigma, (s_A, s_B, 0), \delta_{A \times B}, F_A \times S_B \times \{0\})$
- $(p', q', j) \in \delta_{A \times B}((p, q, i), a)$ pro všechna
 - $p' \in \delta_A(p, a)$
 - $q' \in \delta_B(q, a)$
 - $j = (i + 1) \bmod 2$ pokud $(i = 0 \wedge p \in F_A) \vee (i = 1 \wedge q \in F_B)$
 - $j = i$ v ostatních případech

- Konstrukce $A \times B$ pro případ, že $F_A = S_A$

- $A \times B = (S_A \times S_B, \Sigma, (s_A, s_B), \delta_{A \times B}, S_A \times F_B)$
- $(p', q') \in \delta_{A \times B}((p, q), a)$ pro všechna
 - $p' \in \delta_A(p, a)$
 - $q' \in \delta_B(q, a)$

Nechť

- $L1 = \{w \in \{a, b, c\}^\omega \mid a \in \text{inf}(w)\}$
- $L2 = \{w \in \{a, b, c\}^\omega \mid \text{inf}(w) = \{b\}\}$
- $L3 = L1 \cap L2$

Najděte Büchi automaty pro $L1$, $L2$ a $L3$.

Tvrzení

- Pro každou LTL formuli φ platí: $co-L(A_\varphi) = L(A_{\neg\varphi})$.

Redukce $M \models \varphi$ na problém prázdnosti $L(A_{sys} \times A_{\neg\varphi})$

- $M \models \varphi \iff L_{sys} \subseteq L_\varphi$
- $M \models \varphi \iff L(A_{sys}) \subseteq L(A_\varphi)$
- $M \models \varphi \iff L(A_{sys}) \cap co-L(A_\varphi) = \emptyset$
- $M \models \varphi \iff L(A_{sys}) \cap L(A_{\neg\varphi}) = \emptyset$
- $M \models \varphi \iff L(A_{sys} \times A_{\neg\varphi}) = \emptyset$

Tvrzení

- Büchi automat $A = (S, \Sigma, s_0, \delta, F)$ akceptuje neprázdný jazyk právě když existují stav $s \in F$ a slova $w_1, w_2 \in \Sigma^*$ taková, že $s \in \hat{\delta}(s_0, w_1)$ a $s \in \hat{\delta}(s, w_2)$.
- Tj. graf Büchi automatu obsahuje dosažitelný akceptující cyklus (cyklus přes nějaký akceptující stav).

Rozhodovací procedura pro problém $M \models \varphi$?

- Zkonstruuje se produktový automat $(A_{\text{sys}} \times A_{\neg\varphi})$.
- Graf produktového automatu se prověří na přítomnost akceptujících cyklů.
- Obsahuje-li graf akceptující cyklus, pak $M \not\models \varphi$.
- Neobsahuje-li graf akceptující cyklus, pak $M \models \varphi$.

Vstup algoritmu

- Produktový automat ve formě tří následujících funkcí
 - $F_init()$ — Vrací iniciální stav automatu.
 - $F_succs(s)$ — Pro daný stav vrací jeho přímé následníky.
 - $Accepting(s)$ — O stavu řekne, zda je či není akceptující.

Výstup algoritmu

- Přítomen / Nepřítomen
- Protipříklad.

Algoritmus

- Využívá vnořené prohledávání do hloubky – Nested DFS.
- Vnější procedura detekuje akceptující stavy, vnitřní procedura testuje, zda akceptující stav je dosažitelný ze sebe sama (leží na cyklu).