Aula 6: Hennessy-Milner Logic

Interaction & Concurrency Course Unit: Reactive Systems Module

April 26, 2023

Recommended reading

Chapter 5 of Aceto et al. 2007.

Concepts introduced and discussed:

- propositional modal logic: sintax and Kripke semantics,
- model, frame and valuation,
- satisfaction relation for a model and a world,
- formula satisfiable in a model,
- formula globally satisfied in a model,
- formula valid,
- modal logics over LTS,
- example: a process logic - Hennessy-Milner logic (HML),
- sintax and semantics of HML,
- examples of formulas satisfied in a state of a LTS,
- examples of LTS that satisfy simultaneously a set of formulas, in a state.

Some relevant definitions and examples (from Aceto et al. 2007):

- def. 5.1 (syntax of HML);
- intuitive semantics (pp. 103);
- semantics of HML: definition of satisfiability relation relating processes to formulas by structural induction on formulas (pp. 108).

Hennessy-Milner Logic

Syntax of the Formulae ($a \in Act$)

$$F, G \ ::= \ tt \ | \ ff \ | \ F \wedge G \ | \ F \vee G \ | \ \langle a \rangle F \ | \ [a]F$$

- [ ] → necessidade
  - `[a]F` → todos os sucessores `a` devem satisfazer `F`
- < > → possibilidade
  - `⟨a⟩F` → existe pelo menos um sucessor `a` que satisfaz `F`

Semântica Denotacional

For a formula F let $[\![F]\!] \subseteq Proc$ contain all states that satisfy F.

Denotational Semantics: $[\![\_]\!] : Formulae \rightarrow 2^{Proc}$

- $[\![tt]\!] = Proc$ and $[\![ff]\!] = \emptyset$
- $[\![F \vee G]\!] = [\![F]\!] \cup [\![G]\!]$
- $[\![F \wedge G]\!] = [\![F]\!] \cap [\![G]\!]$
- $[\![\langle a \rangle F]\!] = \langle \cdot a \cdot \rangle [\![F]\!]$
- $[\![[a] F]\!] = [\cdot a \cdot][\![F]\!]$

where $\langle \cdot a \cdot \rangle, [\cdot a \cdot] : 2^{(Proc)} \rightarrow 2^{(Proc)}$ are defined by

$$\langle \cdot a \cdot \rangle S = \{p \in Proc \mid \exists p'.\ p \xrightarrow{a} p' \text{ and } p' \in S\}$$

$$[\cdot a \cdot] S = \{p \in Proc \mid \forall p'.\ p \xrightarrow{a} p' \implies p' \in S\}.$$

$< \cdot a \cdot > S$ = estados a partir dos quais é possível chegar a S

$[\cdot a \cdot] S$ = todos os estados de onde não parte nenhuma transição com a + todas as transições por a de um dado estado têm de chegar a um estado que esteja em S

conjunto de chegada

Exercises suggested (from Aceto et al. 2007):

- Exercise 5.2; ✓ pg 106
- Exercise 5.3.1; ✓ pg 107
- Exercise 5.4; ✓ pg 107
- Exercise 5.5. ✓ pg 108

(to be completed)

References

Aceto, Luca et al. (2007). *Reactive Systems - Modelling, Specification and Verification*. Cambridge University Press.

Dê fórmulas que expressam os seguintes requisitos de linguagem natural:

1 • o processo está disposto a beber café e chá agora;

2 • o processo está disposto a beber café, mas não chá agora;

3 • o processo pode sempre beber chá imediatamente após ter bebido dois cafés seguidos.

3. O que as fórmulas ⟨a⟩ff e [a]tt expressam?

1. $\langle \text{café} \rangle \langle \text{chá} \rangle \text{ true}$

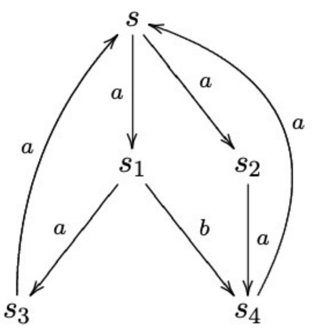
2. $\langle \text{café} \rangle \text{ true} \quad \langle \text{chá} \rangle \text{ false}$

3. $\langle \text{café} \rangle \langle \text{café} \rangle \langle \text{chá} \rangle \text{ true}$

illp -

$\langle a \rangle ff$ → existe um estado com nenhuma transição por a.

$[a]\, tt$ → todas as transições de um estado são por a.



$< \cdot a \cdot > S$ = estados a partir dos quais é possível chegar a S

$[\cdot a \cdot] S$ = todos os estados de onde não parte nenhuma transição com a + todas as transições por a de um dado estado têm de chegar a um estado que esteja em S

$S \models \langle a \rangle tt$ ✓

$S \models \langle b \rangle tt$ ✗

$S \models [a]\, ff$ ✗

$S \models [b]\, ff$ ✓

$S \models [a][b]\, tt$ ✗

$S \models \langle a \rangle \langle b \rangle tt$ ✓

$S \models [a] \langle a \rangle [a][b]\, ff$ ✓

$S \models \langle a \rangle (\langle a \rangle \text{true} \wedge \langle b \rangle tt)$ ✓

$S \models [a] (\langle a \rangle \text{true} \vee \langle b \rangle tt)$ ✓

$S \models \langle a \rangle ([b][a] ff \wedge \langle b \rangle tt)$ ✗

$S \models \langle a \rangle ([a](\langle a \rangle tt \wedge [b] ff) \wedge \langle b \rangle ff)$ ✓

$tt \to \text{PROC}$

$ff \to \emptyset$

(a) $s \models \langle a \rangle tt$

$\Leftrightarrow$ $s \models \langle a \rangle PROC = \{s, s_1, s_2, s_3, s_4\}$

$\Leftrightarrow$ $s \models \{s, s_1, s_2, s_3, s_4\}$

(b) $s \models \langle b \rangle tt$

$\Leftrightarrow$ $s \models \langle b \rangle PROC$

$\Leftrightarrow$ $s \not\models \{s_1\}$ pcq $s \notin \{s_1\}$

(c) $s \models [a]\, ff$

Rever

$\Leftrightarrow$ $s \models [a]\, \emptyset$

$\Leftrightarrow$ $s \not\models \emptyset$ pcq $s \notin \emptyset$

(d) $s \models [b]\, ff$

$\Leftrightarrow$ $s \models [b]\, \emptyset$

$\Leftrightarrow$ $s \models \{s, s_2, s_3, s_4\} \ni s$

(e) $s \models [a] \langle b \rangle tt$

$\Leftrightarrow$ $s \models [a] \langle b \rangle PROC$

$\Leftrightarrow$ $s \models [a] \{s_1\}$

$\Leftrightarrow$ $s \not\models \emptyset$ pcq $s \notin \emptyset$

(f) $s \models \langle a \rangle \langle b \rangle tt$

$\Leftrightarrow$ $s \models \langle a \rangle \langle b \rangle PROC$

$\Leftrightarrow$ $s \models \langle a \rangle \{s_1\}$ $\Leftrightarrow$ $s \models \{s\} \ni s$

(g) $s \models [a]\langle a\rangle [a][b]\, ff$

$\Leftrightarrow s \models [a]\langle a\rangle[a][b]\, \emptyset$

$\Leftrightarrow s \models [a]\langle a\rangle[a]\{s, s_2, s_3, s_4\}$

$\Leftrightarrow s \models [a]\langle a\rangle\{s_1, s_2, s_3, s_4\}$

$\Leftrightarrow s \models [a]\{s, s_1, s_2\}$

$\Leftrightarrow s \models \{s, s_3, s_4\} \not\ni s$

(h) $s \models \langle a\rangle(\langle a\rangle tt \wedge \langle b\rangle tt)$

$\Leftrightarrow s \models \langle a\rangle(\langle a\rangle PROC \cap \langle b\rangle PROC)$

$\Leftrightarrow s \models \langle a\rangle(PROC \cap \{s_1\})$

$\Leftrightarrow s \models \langle a\rangle\{s_1\}$

$\Leftrightarrow s \models \{s\} \ni s$

Semântica denotacional

(a) $[\![[a][b]\, ff]\!]$

$= [\cdot a \cdot][\cdot b \cdot]\, \emptyset$

$= [\cdot a]\{s, s_2, s_3, s_4\}$

$= \{s_3, s_4, s, s_1, s_2\}$

Exercise 5.4 *Consider an everlasting clock whose behaviour is defined thus:*

$$Clock \stackrel{def}{=} tick.Clock \ .$$

Prove that the process Clock satisfies the formula

$$[tick](\langle tick\rangle tt \wedge [tock]ff) \ .$$

$$\text{CLOCK} \circlearrowleft \text{TICK}$$

$\text{CLOCK} \models [\text{TICK}](\langle \text{TICK}\rangle tt \wedge [\text{TOCK}] ff)$

$\Leftrightarrow \text{CLOCK} \models [\text{TICK}](\langle \text{TICK}\rangle \text{PROC} \cap [\text{TOCK}] \emptyset)$ → NÃO SEI SE POSSO FAZER ESTA SUBSTITUIÇÃO

$\Leftrightarrow \text{CLOCK} \models [\text{TICK}](\{\text{CLOCK}\} \cap \{\text{CLOCK}\})$

$\Leftrightarrow \text{CLOCK} \models \{\text{CLOCK}\} \ni \text{CLOCK}$

Show also that, for each $n \geq 0$*, the process Clock satisfies the formula*

$$\underbrace{\langle tick\rangle \cdots \langle tick\rangle}_{n\text{-times}} tt \ .$$

$\text{CLOCK} \models \langle \text{TICK}\rangle \ldots \langle \text{TICK}\rangle tt$

$\Leftrightarrow \text{CLOCK} \models \underbrace{\langle \text{TICK}\rangle \ldots \langle \text{TICK}\rangle}_{n\times} \text{PROC}$

$\Leftrightarrow \text{CLOCK} \models \underbrace{\langle \text{TICK}\rangle \ldots \langle \text{TICK}\rangle}_{(n-1)\times} \text{PROC}$

$\Leftrightarrow \ldots$

$\Leftrightarrow \text{CLOCK} \models \text{PROC} \ni \text{CLOCK}$

Exercise 5.5 (Mandatory) *Find a formula in $\mathcal{M}$ that is satisfied by $a.b.\mathbf{0} + a.c.\mathbf{0}$, but not by $a.(b.\mathbf{0} + c.\mathbf{0})$.*

Find a formula in $\mathcal{M}$ that is satisfied by $a.(b.c.\mathbf{0} + b.d.\mathbf{0})$, but not by $a.b.c.\mathbf{0} + a.b.d.\mathbf{0}$. ♦

$s \models a.b.Nil + a.c.Nil \not\Dashv t$

$s \not\models a.(b.Nil + c.Nil) \Dashv t$

Fórmula: $[a][b]\,\mathrm{false}$ $\Dashv s$, $\not\Dashv t$

$s \models a.(b.c.Nil + b.d.Nil) \not\Dashv t$

$s \not\models a.b.c.Nil + a.b.d.Nil \Dashv t$

illp.

Fórmula: $[a]\langle b\rangle\langle c\rangle\,\mathrm{true}$ $\Dashv s$, $\not\Dashv t$

Fórmula que os distingue: $\langle a\rangle[b]\langle c\rangle\,\mathrm{true}$ $\Dashv t$; $\not\Dashv s$

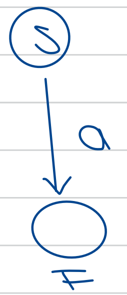
(a) $\langle a\rangle (\langle b\rangle \langle c\rangle true \wedge \langle c\rangle true)$ (*)

(b) $\langle a\rangle \langle b\rangle ([a] false \wedge [b] false \wedge [c] false)$

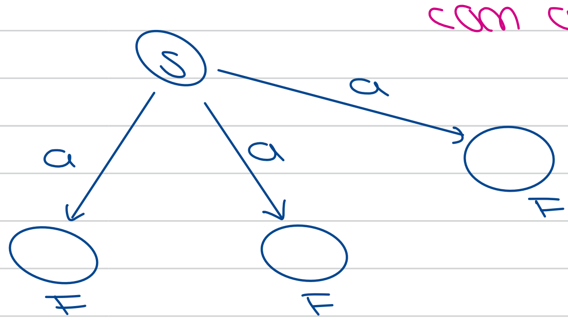
(c) $[a] \langle b\rangle ([c] false \wedge \langle a\rangle true)$

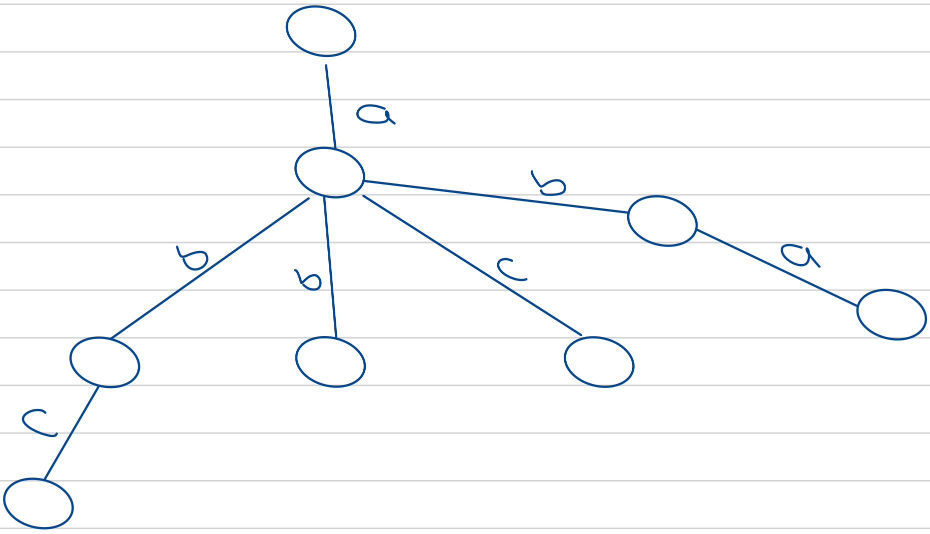
não há nenhuma trans com a, b, c

$S \models \langle a\rangle F$



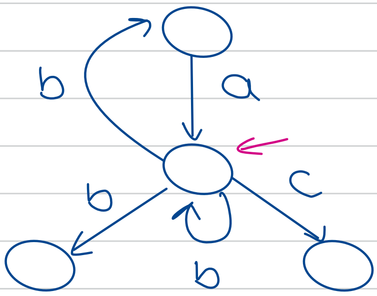
$S \models [a] F$



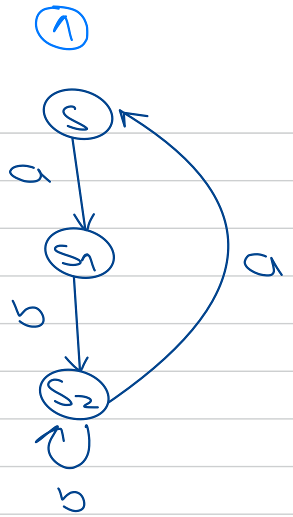


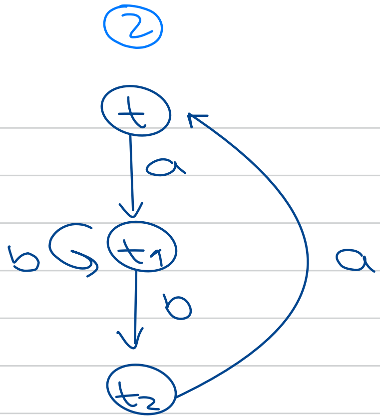
→ construção de (a) + (b) + (c)

Forma mais compacta

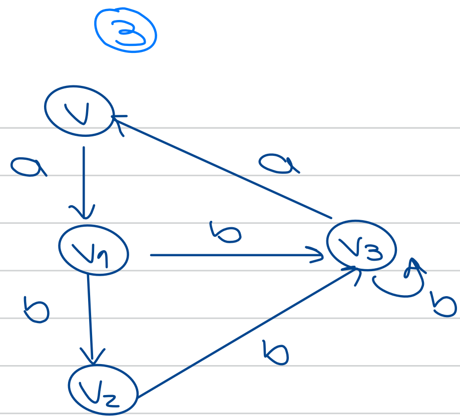


começamos a validar a partir daqui (*)





$t_1 \xrightarrow{b} t_1$ ✓
$t_1 \xrightarrow{b} t_2$ ✗
$t_1 \xrightarrow{a} t_1$
$t_2 \xrightarrow{b}$ ✗



$v_1 \xrightarrow{b} v_2$ ✓
$v_1 \xrightarrow{b} v_3$ ✓

Exercício:

$F = \langle a \rangle [b] \langle b \rangle \, true$ — analisar todas as trans.

Em qual dos sistemas a fórmula é válida

$S \overset{?}{\models} F$ ✓ $\quad T \overset{?}{\models} F$ ✗ $\quad V \overset{?}{\models} F$ ✓

$S \models \langle a \rangle [b] \langle b \rangle \, true$

$\exists s' : s \xrightarrow{a} s' \wedge (s' \models [b] \langle b \rangle \, true)$

$s' \models [b] \langle b \rangle \, true$

$\exists s'' : s' \xrightarrow{b} s'' \wedge (s'' \models \langle b \rangle \, true)$

Mt imp.

Queremos encontrar uma fórmula que permita distinguir ① e ③.

$S \models \langle a \rangle [b] \langle a \rangle$
$V \not\models \langle a \rangle [b] \langle a \rangle$

Verificar se a expressão dada se verifica em ①, ② e ③:

$\langle a \rangle \langle b \rangle (\langle a \rangle \, true \wedge \langle b \rangle \, true) = F$

$S \models F$
$T \not\models F$ pq quando tenho caminhos por a não tem por b e vice-versa.

$t_1 \overset{?}{\models} \mathcal{Z}$

$t_1 \xrightarrow{b} t_1 \xrightarrow{a} \times$ (falha)
$t_1 \xrightarrow{b} t_2 \xrightarrow{a} \top$

$V \models F$

e se tivessemos:

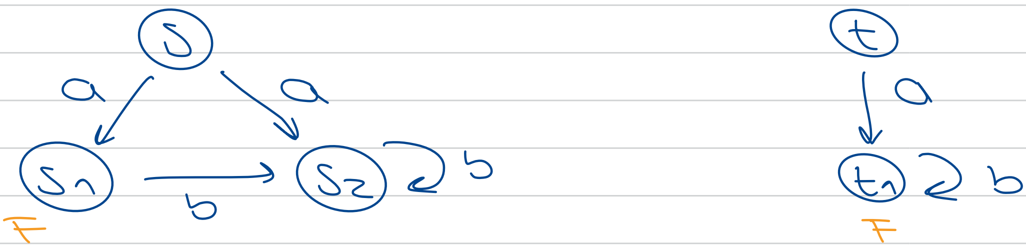
$\langle a \rangle [b] (\langle a \rangle true \wedge \langle b \rangle true) = F$

$S \models F$
$T \not\models F$
$V \not\models F$

$\Downarrow$

outra explicação
Aula 7

Exemplo:



$$\underbrace{\langle \cdot a \cdot \rangle \{s_1, t_1\}}_{\text{é possível chegar a } \underline{s_1} \text{ e } \underline{t_1} \text{ por } \underline{a}\text{?}} = \underbrace{\{s, t\}}_{\text{sim, por } \underline{s} \text{ e } \underline{t}}$$

$$[\cdot a \cdot]\{s_1, t_1\} = \{s_1, s_2, t, t_1\}$$

$$\langle \cdot b \cdot \rangle \{s_1, t_1\} = \{t_1\}$$

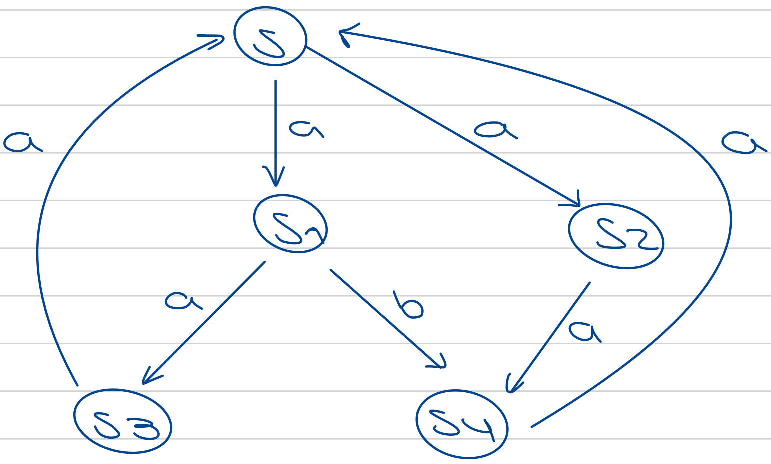
$$[\cdot b \cdot]\{s_1, t_1\} = \{s, t, \cancel{s_1}, t_1\}$$

↪ $s_1 \xrightarrow{b} s_2$ mas $s_2 \notin \{\ldots\}$

↓

todos os estados de onde não parte nenhum b ⊕ trans. por b que estão no conjunto $\{\ldots\}$

Exercício:



(a) $[\![[a][b]\, ff]\!]$ $\{\}$

(b) $[\![\langle a \rangle (\langle a \rangle\, true \wedge \langle b \rangle\, true)]\!]$

(c) $[\![[a][a][b]\, false]\!]$

(d) $[\![[a] (\langle a \rangle\, true \vee \langle b \rangle\, true)]\!]$

(a) $[\cdot a \cdot] ([\![[b]\, ff]\!])$

$= [\cdot a \cdot] ([\cdot b \cdot] [\![ff]\!])$

$= [\cdot a \cdot] ([\cdot b \cdot]\, \emptyset)$

$= [\cdot a \cdot] \{s, s_2, s_3, s_4\}$ Todos os estados que não têm trans. por b.

$= \{s_2, s_3, s_4, s_1\}$

↳ pq $s_1 \xrightarrow{a} s_3$ e $s_3 \in \{\dots\}$

$s \xrightarrow{a} s_1$ mas $s_1 \notin \{\dots\}$

(b) $[\![\langle a \rangle (\langle a \rangle\, true \wedge \langle b \rangle\, true)]\!]$

$= \langle \cdot a \cdot \rangle (\langle \cdot a \cdot \rangle \{s, s_1, s_2, s_3, s_4\} \cap \langle \cdot b \cdot \rangle\, true)$

$= \langle \cdot a \cdot \rangle (\{s, s_1, s_2, s_3, s_4\} \cap \{s_1\})$

$= \langle \cdot a \cdot \rangle \{s_1\}$

$= \{s\}$

(c) $[\![\langle a] \, [a] \, \{b\} \, \mathit{false}]\!]$

$= \langle \cdot a \cdot] \, (\langle\, [\, a \cdot] (\langle [\cdot b \cdot] \; \emptyset \,))$

$= [\cdot a \cdot] \, (\langle\, [\cdot a \cdot] \, \{ s, s_2, s_3, s_4 \})$

$= [\cdot a \cdot] \, \{ s_1, s_2, s_3, s_4 \}$

$= \{ s_1, s_2, s \}$

Todas as trans:

✗ $s \xrightarrow{a} s_1$ mas $s_1 \notin$

$s \xrightarrow{a} s_2$

✓ $s_1 \xrightarrow{a} s_3$

✓ $s_2 \xrightarrow{a} s_4$

✓ $s_3 \xrightarrow{a} s$

$s_4 \xrightarrow{a}$