Aulas 8 e 9: Hennessy-Milner Logic with recursion

Interaction & Concurrency Course Unit: Reactive Systems Module

May 5, 2023

Recommended reading

Chapter 6 of Aceto et al. 2007 and chapter 6 of Groote and Mousavi 2014.

Concepts introduced and discussed:

- syntax and semantics of HML with recursion,
- greatest fixed points and least fixed points,
- invariance properties, safety and liveness properties, temporal properties,
- game characterization for HML with recursion.

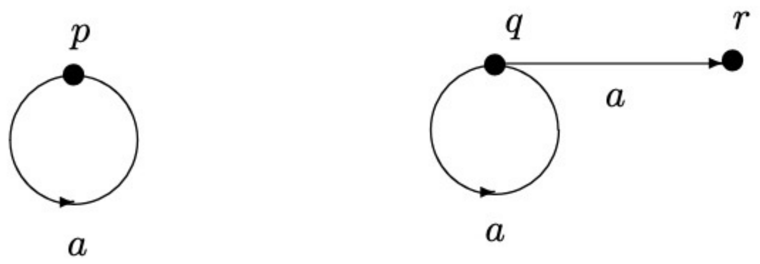
Some relevant definitions, examples, theorems (from Aceto et al. 2007):

- example 6.1,
- abstract syntax of HML with recursion pp. 109,
- section 6.1 (examples of recursive properties)
- def. 6.1 (semantics of HML with recursion),
- example 6.2,
- theorem 6.1,
- theorem 6.3 (game characterization), → pg 129 | 130 | 131 132 | 133 VER ✓
- examples 6.3-6.7

Exercises suggested (from Aceto et al. 2007):

- ~~exercise 6.4;~~
- exercise 6.6; ✓ pg 126
- exercise 6.7; ✓ pg 134

Não é difícil encontrar uma fórmula HML que p satisfaça e q não. De fato, depois de realizar uma a-ação, p sempre poderá realizar outra, enquanto q pode não conseguir. Isso pode ser capturado formalmente em HML da seguinte maneira:



$p \models [a]\langle a\rangle tt$

$q \not\models [a]\langle a\rangle tt$

$$r = r_0 \xrightarrow{a} r_1 \xrightarrow{a} r_2 \xrightarrow{a} r_3 \cdots r_{n-1} \xrightarrow{a} r_n \quad (n \geq 0) \ .$$

No matter how we choose a non-negative integer n, there is an HML formula that distinguishes the processes p and q. In fact, we have that

$$\begin{array}{lll} p & \models & [a]^{n+1}\langle a\rangle tt \quad \text{but} \\ q & \not\models & [a]^{n+1}\langle a\rangle tt \ , \end{array}$$

Disto temos:

$$Inv(\langle a\rangle tt) = \langle a\rangle tt \wedge [a]\langle a\rangle tt \wedge [a][a]\langle a\rangle tt \wedge \cdots = \bigwedge_{i\geq 0} [a]^i \langle a\rangle tt \ .$$

$$Pos([a]ff) = [a]ff \vee \langle a\rangle[a]ff \vee \langle a\rangle\langle a\rangle[a]ff \vee \cdots = \bigvee_{i\geq 0} \langle a\rangle^i [a]ff \ ,$$

$O_F(S)$ é um conjunto de processos que satisfazem F

$$X \stackrel{\max}{=} F \wedge ([\mathsf{Act}]ff \vee \langle \mathsf{Act}\rangle X) \ .$$

$$\underbrace{Y \stackrel{\min}{=} F \vee (\langle \mathsf{Act}\rangle tt \wedge [\mathsf{Act}]Y) \ .}_{\text{Pontos Fixos}}$$

$$\begin{array}{rcl} O_X(S) & = & S \\ O_{tt}(S) & = & Proc \\ O_{ff}(S) & = & \emptyset \\ O_{F_1 \wedge F_2}(S) & = & O_{F_1}(S) \cap O_{F_2}(S) \\ O_{F_1 \vee F_2}(S) & = & O_{F_1}(S) \cup O_{F_2}(S) \\ O_{\langle a\rangle F}(S) & = & \langle \cdot a \cdot \rangle O_F(S) \\ O_{[a]F}(S) & = & [\cdot a \cdot] O_F(S) \end{array}$$

Atacante / Defensor

Regras do jogo:

1. $(s, F_1 \wedge F_2)$ ou $(s, [a]F) \rightarrow$ Atacante
2. $(s, F_1 \vee F_2)$ ou $(s, \langle a \rangle F) \rightarrow$ Defensor
3. $(s, X) \rightarrow (s, F_X)$
4. (s, tt) ou (s, ff)
5. Atacante $(s, [a]f)$ $s \overset{a}{\nrightarrow}$
6. Defensor $(s, \langle a \rangle f)$ $s \overset{a}{\nrightarrow}$
7. jogo infinito

$\}$ jogo termina

Mt imp.

Winner:

1. A: O jogo termina (s, false) ou D bloqueado
2. D: O jogo termina (s, true) ou A bloqueado
3. A: jogo infinito $X \overset{min}{=} F_X$
4. D: jogo infinito $X \overset{max}{=} F_X$

Próximas Configurações

A → Atacante $s \not\models F$
D → Defensor $s \models F$

configurações (s, F)

Próximas configurações:

(s, true), (s, false) sem conf. seguinte

$(s, F_1 \wedge F_2) \rightarrow (s, F_1)$ e (s, F_2)

$(s, F_1 \vee F_2) \rightarrow (s, F_1)$ e (s, F_2)

$(s, [a]F)$ ou $(s, \langle a \rangle F)$ (s', F) todos os $s \xrightarrow{a} s'$

$(s, X) \rightarrow (s, F_X)$ $X = F_X$

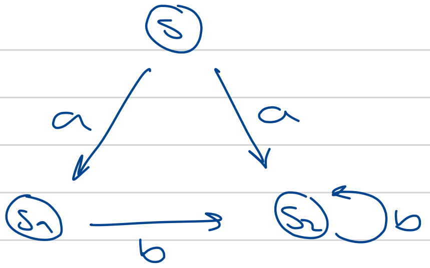
Pontos fixos

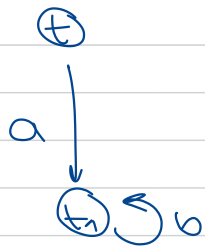
Max → O atacante tem que mostrar que a fórmula falha num nº finito de passos
- se chegarmos a um jogo infinito o atacante perde

Min → O defensor tem que encontrar uma estratégia finita.

Se uma dada configuração já apareceu anteriormente, então estamos perante um jogo infinito

$\langle ACT \rangle$ → por a ou b ou c
$[ACT]$ → por a e b e c





$\langle a \rangle \emptyset = \emptyset$

$[a]\, Proc = Proc$

$$Y \overset{min}{=} \langle b \rangle tt \vee \langle \{a,b\} \rangle Y$$

↳ $\langle a \rangle F \cup \langle b \rangle F$

$[\![Y]\!] = [\![F_Y]\!]$

$Y = \emptyset$

$F_Y = \langle b \rangle Proc \cup \langle \{a,b\} \rangle \emptyset$

$= \{s_1, s_2, t_1\} \cup \emptyset$

$= \{s_1, s_2, t_1\}$

$Y = \{s_1, s_2, t_1\}$

$F_Y = \langle b \rangle Proc \cup \langle \{a,b\} \rangle \{s_1, s_2, t_1\}$

$= \{s_1, s_2, t_1\} \cup \{s, t, s_1, s_2, t_1\}$

$= Proc$

$Y = Proc$

$F_Y = \langle b \rangle Proc \cup \langle \{a,b\} \rangle Proc$

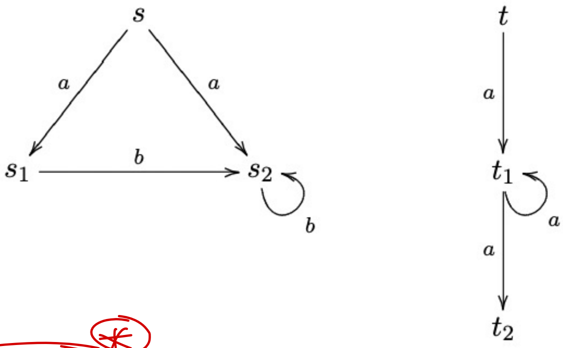
$= \{s_1, s_2, t_1\} \cup Proc$

$= Proc$

$[\![Y]\!] \overset{min}{=} Proc$

Proc é o menor ponto fixo.

Exercise 6.7 *Consider the labelled transition system*



Use the game characterization for HML with recursion to show that

1. s_1 *satisfies the formula*

$$X \overset{\max}{=} \langle b \rangle \mathit{tt} \wedge [b] X \ ;$$

2. s *satisfies the formula*

$$Y \overset{\min}{=} \langle b \rangle \mathit{tt} \vee \langle \{a, b\} \rangle Y \ ,$$

but t *does not.*

Find a recursively defined property that t *satisfies and argue that it does so using the game characterization of satisfaction presented above.* ♦

$[\![X]\!] = [\![F_{max}]\!]$

$X = Proc$

$F_{max} = \langle b \rangle tt \cap [b] Proc$

$= \langle b \rangle Proc \cap [b] Proc$

$= \{s_1, s_2\} \cap \{s, t, t_1, t_2, s_1, s_2\}$

$= \{s_1, s_2\}$

$X = \{s_1, s_2\}$

$F_{max} = \langle b \rangle Proc \cap [b] \{s_1, s_2\}$

$= \{s_1, s_2\} \cap \{s, t, t_1, t_2, s_1, s_2\}$

$= \{s_1, s_2\}$

$s_1 \models F_{max}$ pq $s_1 \in \{s_1, s_2\}$

(*)

GAME

$(s_1, X) \to (s_1, \langle b\rangle tt \wedge [b]X)$

(1) $\xrightarrow{A} (s_1, \langle b\rangle tt)$

$\xrightarrow{D} (s_2, tt) \longrightarrow$ D ganha

(2) $\xrightarrow{A} (s_1, [b]X)$

$\xrightarrow{A} (s_2, X) \longrightarrow$ jogo infinito

Como $X \overset{max}{=} FX$ o defensor ganha e portanto temos $s_1 \models X$.

$S \models Y$
Estratégia vencedora Defensor

$(s, Y) \to (s, \langle b\rangle tt \vee \langle a,b\rangle Y)$

(1) $\cancel{\xrightarrow{D} (s, \langle b\rangle tt)}$

$\cancel{\xrightarrow{D} s \not\xrightarrow{b}} \longrightarrow$ D perde ; A ganha

(2) $\xrightarrow{D} (s, \langle a,b\rangle Y)$

$\xrightarrow{D} (s, \langle a\rangle Y \vee \langle b\rangle Y) \longrightarrow$ A ganha

(1) $\xrightarrow{D} (s, \langle a\rangle Y)$

$\xrightarrow{D} (s_1, Y)$

$\longrightarrow (s_1, \langle b\rangle tt \vee \langle a,b\rangle Y)$

$\xrightarrow{D} (s_1, \langle b\rangle tt)$

$\xrightarrow{D} (s_2, tt) \longrightarrow$ D ganha $\therefore S \models Y$

(2) $\cancel{\xrightarrow{D} (s, \langle b\rangle Y)}$

$\cancel{\xrightarrow{D} s \not\xrightarrow{b}} \longrightarrow$ A ganha

Simplificar: $s \models Y$ → Estratégia vencedora defensor

$(s, \langle b \rangle tt \vee \langle a,b \rangle Y)$

$\xrightarrow{D} (s, \langle a,b \rangle Y)$

$\rightarrow (s, \langle a \rangle Y \vee \langle b \rangle Y)$

$\xrightarrow{D} (s, \langle a \rangle Y)$

1 $\xrightarrow{D} (s_1, Y)$

$\rightarrow (s_1, \langle b \rangle tt \vee \langle a,b \rangle Y)$

$\xrightarrow{D} (s_1, \langle b \rangle tt)$

$\xrightarrow{D} (s_2, tt)$ → D ganha

2 $\xrightarrow{D} (s_2, Y)$

$\rightarrow (s_2, \langle b \rangle tt \vee \langle a,b \rangle Y)$

$\xrightarrow{D} (s_2, \langle b \rangle tt)$

$\xrightarrow{D} (s_2, tt)$ → D ganha

$t \nvDash Y$ onde $Y \stackrel{\min}{=}$ → Defensor tenta encontrar estratégia finita

$(t, Y) \to (t, \langle b \rangle tt \vee \langle\{a,b\}\rangle Y)$

$\xrightarrow{D} (t, \langle a \rangle Y \vee \langle b \rangle Y)$

$\xrightarrow{D} (t, \langle a \rangle Y)$

$\xrightarrow{D} (t_1, Y)$

$\to (t_1, \langle b \rangle tt \vee \langle\{a,b\}\rangle Y)$

$\xrightarrow{D} (t_1, \langle a \rangle Y \vee \langle b \rangle Y)$

$\xrightarrow{D} (t_1, \langle a \rangle Y)$

1. $\xrightarrow{D} (t_1, Y)$ → jogo infinito, A ganha

2. $\xrightarrow{D} (t_2, Y)$

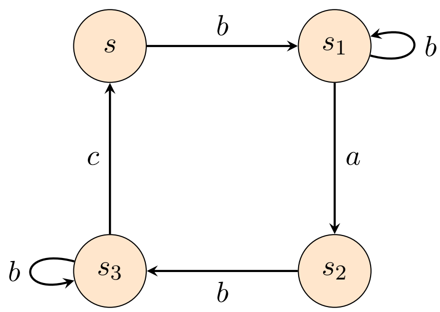
$\to (t_2, \langle a \rangle Y \vee \langle b \rangle Y)$

$\xrightarrow{D}$ $t_2 \not\xrightarrow{a}$ → D bloqueado
$t_2 \not\xrightarrow{b}$ A ganha

$\therefore$ $t \nvDash Y$

Other exercises suggested

- In the context of the examples of section 6.4.1 of Aceto et al. 2007, use Theorem 6.1 to compute the set of processes that satisfy each of the properties mentioned in examples 6.3-6.7.
- Repeat the exercise for the LTS and formulas of exercise 6.7.
- Consider the following labelled transition system.



Using the game characterization for recursive Hennessy-Milner formulae decide whether the following claims are true or false and discuss what properties the formulae describe:

(a) – $s \overset{?}{\models} X$ where $X \overset{min}{=} \langle c\rangle true \vee \langle Act\rangle X$

(b) – $s \overset{?}{\models} X$ where $X \overset{min}{=} \langle c\rangle true \vee [Act]X$

(c) – $s \overset{?}{\models} X$ where $X \overset{max}{=} \langle b\rangle X$

(d) – $s \overset{?}{\models} X$ where $X \overset{max}{=} \langle b\rangle true \wedge \underbrace{[a]X \wedge [b]X}_{\text{Não servem Atacante perde logo}}$

min
começa 2 4

max
começa PROC

References

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

Groote, Jan and Mohammad Mousavi (2014). *Modelling and Analysis of Communicating Systems*. The MIT Press.

↗ Defensor tem que encontrar uma estratégia finita.

$s \models X$ onde $X \overset{min}{=} \langle c \rangle true \vee \langle Act \rangle X$

$(s, X) \longrightarrow (s, \langle c \rangle true \vee \langle Act \rangle X)$

$\xrightarrow{D} (s, \langle Act \rangle X)$

$\xrightarrow{D} (s_1, X)$

$\longrightarrow (s_1, \langle c \rangle true \vee \langle Act \rangle X$

$\xrightarrow{D} (s_1, \langle Act \rangle X)$

① $(s_1, X) \longrightarrow$ jogo infinito; A ganha

② $\xrightarrow{D} (s_2, X)$

$\longrightarrow (s_2, \langle c \rangle true \vee \langle Act \rangle X)$

$\xrightarrow{D} (s_3, X)$

$\longrightarrow (s_3, \langle c \rangle true \vee \langle Act \rangle X)$

$\xrightarrow{D} (s_3, \langle c \rangle true)$

$\xrightarrow{D} (s, true) \longrightarrow$ D ganha

$\therefore s \models X$

$s \models X$ onde $X \overset{max}{=} \langle b \rangle X$

$(s, \langle b \rangle X) \xrightarrow{D} (s_1, X)$

$\longrightarrow (s_1, \langle b \rangle X)$

$\longrightarrow (s_1, X) \longrightarrow$ jogo infinito D ganha

(b) $s \models X$ onde $X \stackrel{min}{=} \langle c \rangle true \vee [Act] X$

Encontrar ponto fixo

$X \stackrel{min}{=} \langle c \rangle true \vee [Act] X$

$[\![X]\!] = \{\}$

$[\![X]\!] = [\![F_X]\!] = \langle \cdot c \cdot \rangle PROC \cup [\cdot Act \cdot] \emptyset$

$= \{s3\} \cup \emptyset$

$= \{s3\}$

$[\![X]\!] = \{s3\}$

$[\![X]\!] = [\![F_X]\!] = \langle \cdot c \cdot \rangle PROC \cup [\cdot Act \cdot] \{s3\}$

$= \{s3\} \cup \{s2\}$

$= \{s2, s3\}$

$[\![X]\!] = \{s2, s3\}$

$[\![X]\!] = [\![F_X]\!] = \langle \cdot c \cdot \rangle PROC \cup [\cdot Act \cdot] \{s2, s3\}$

$= \{s3\} \cup \{s2\}$

$= \{s2, s3\}$

Ponto fixo $= \{s2, s3\}$

Mas como $s \notin \{s2, s3\}$ temos que $s \not\models X$

imp.

Game

$(s, X) \longrightarrow (s, \langle C \rangle true \vee [Act] X)$

$\xrightarrow{D} (s, [Act] X)$

$\xrightarrow{A} (s_1, X)$

$\longrightarrow (s_1, \langle C \rangle true \vee [Act] X)$

$\xrightarrow{D} (s_1, [Act] X)$

1 $\xrightarrow{A} (s_1, X) \longrightarrow$ jogo infinito como $X \stackrel{max}{=}$

$\Downarrow$

A ganha

2 $\xrightarrow{A} (s_2, X)$

$\longrightarrow (s_2, \langle C \rangle true \vee [Act] X)$

$\xrightarrow{D} (s_2, [Act] X)$

$\xrightarrow{A} (s_3, X)$

$\longrightarrow (s_3, \langle C \rangle true \vee [Act] X)$

$\xrightarrow{D} (s_3, \langle C \rangle true)$

$\xrightarrow{D} (s, true)$

Temos então que, $s \models X$

(d) $s \models X$ onde $X \stackrel{max}{=} \langle b \rangle true \wedge [a]X \wedge [b]X$

$$(s, X) \xrightarrow{A} (s, [b]X)$$
$$\xrightarrow{A} (s_1, X)$$

D ganha — jogo infinito

$$\to (s_1, \langle b \rangle true \wedge \underline{[a]X} \wedge [b]X)$$
$$\xrightarrow{A} (s_1, [a]X)$$
$$\xrightarrow{A} (s_2, X)$$
$$\to (s_2, \langle b \rangle true \wedge [a]X \wedge [b]X)$$
$$\xrightarrow{A} (s_2, [b]X)$$
$$\xrightarrow{A} (s_3, X)$$
$$\to (s_3, \langle b \rangle true \wedge [a]X \wedge [b]X)$$

1. $\xrightarrow{A} (s_3, \langle b \rangle true)$

   $\xrightarrow{D} (s_3, true)$ → D ganha

2. $\xrightarrow{A} (s_3, [a]X)$

   $\xrightarrow{A} s_3 \not\xrightarrow{a}$ → A bloqueado; D ganha

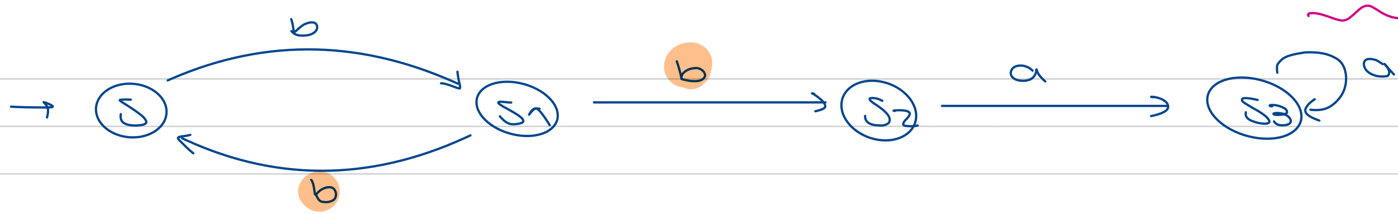
3. $\xrightarrow{A} (s_3, [b]X)$

   $\xrightarrow{A} (s_3, X)$ → jogo infinito; D ganha

$\therefore s \models X$

X é válida em todos os estados

GAME



$s \models [b]\underbrace{(\langle b\rangle [b]\,\text{false} \wedge \langle b\rangle [a]\,\text{false})}_{Y}$

$(s, Y) \xrightarrow{A} (s_1, \langle b\rangle [b]\,\text{false} \wedge \langle b\rangle [a]\,\text{false})$

(1) $\xrightarrow{A} (s_1, \langle b\rangle [b]\,\text{false})$

(1) $\cancel{\xrightarrow{D} (s, [b]\,\text{false})}$

$\cancel{\xrightarrow{A} (s_1, \text{false})}$ → ~~A ganha~~

Estratégia vencedora do defensor → (2) $\xrightarrow{D} (s_2, [b]\,\text{false})$

$\xrightarrow{A} s_2 \not\xrightarrow{b}$ → D ganha

(2) $\xrightarrow{A} (s_1, \langle b\rangle [a]\,\text{false})$

$\xrightarrow{D} (s, [a]\,\text{false})$

$\xrightarrow{A} s \not\xrightarrow{a}$ → D ganha

$s \models X$ onde $X \overset{min}{=} \langle a\rangle tt \vee \langle b\rangle X$

$(s, X) \rightarrow (s, \langle a\rangle tt \vee \langle b\rangle X)$

Estr. vencedora Defensor $\xrightarrow{D} (s, \langle b\rangle X)$ (*)

$\xrightarrow{D} (s_1, X)$

$\rightarrow (s_1, \langle a\rangle tt \vee \langle b\rangle X)$

(*) $\xrightarrow{D} (s_1, \langle b\rangle X)$

$\xrightarrow{D} (s_2, X)$

$\rightarrow (s_2, \langle a\rangle tt \vee \langle b\rangle X)$

$(*) \xrightarrow{D} (s_2, \langle a \rangle tt)$

$\xrightarrow{D} (s_3, tt)$ → D ganha

$s \not\models X$ onde $X \overset{max}{=} \langle b \rangle tt \wedge [b] X$

$(s, X) \xrightarrow{A} (s, [b] X)$ → Estratégia vencedora atacante

$\xrightarrow{A} (s_1, X)$

$\rightarrow (s_1, [b] X)$

$\xrightarrow{A} (s_2, X)$

$\rightarrow (s_2, \langle b \rangle tt)$

$\xrightarrow{D} s_2 \not\xrightarrow{b}$ → A ganha

$s_2 \models X$ onde $X \overset{max}{=} \langle a \rangle tt \wedge [a] X$

↳ Estratégia vencedora do defensor

$(s_2, \langle a \rangle tt \wedge [a] X)$

1 $\xrightarrow{A} (s_2, \langle a \rangle tt)$

$\xrightarrow{D} (s_3, tt)$ → D ganha

2 $\xrightarrow{A} (s_2, [a] X)$

$\xrightarrow{A} (s_3, X)$

$\rightarrow (s_3, \langle a \rangle tt \wedge [a] X)$

$\xrightarrow{A} (s_3, \langle a \rangle tt)$

$\xrightarrow{D} (s_3, tt)$ → D ganha

→ Estratégia vencedora do atacante

$s_1 \not\models X$ onde $X \stackrel{min}{=} \langle a\rangle tt \vee ([b]X \wedge \langle b\rangle tt)$

$(s_1, \langle a\rangle tt \vee ([b]X \wedge \langle b\rangle tt))$

1 $\xrightarrow{D} (s_1, \langle a\rangle tt)$

$\xrightarrow{D} s_1 \not\xrightarrow{a}$ → A ganha

2 $\xrightarrow{D} (s_1, [b]X \wedge \langle b\rangle tt)$

1 $\xrightarrow{A} (s_1, [b]X)$

duas possibilidades

1 $\xrightarrow{A} (s_2, X)$

$\rightarrow (s_2, \langle a\rangle tt \vee ([b]X \wedge \langle b\rangle tt))$

1 $\xrightarrow{D} (s_2, \langle a\rangle tt)$

$\xrightarrow{D} (s_3, tt)$ → D ganha

$\Downarrow$

2 $(\ldots)$

(*) Queremos que ganhe o Atacante

(*) 2 $\xrightarrow{A} (s, X)$

$\rightarrow (s, \langle a\rangle tt \vee ([b]X \wedge \langle b\rangle tt))$

1 $\xrightarrow{D} (s, \langle a\rangle tt)$

$\xrightarrow{D} s \not\xrightarrow{a}$ → A ganha

2 $\xrightarrow{D} (s, [b]X \wedge \langle b\rangle tt)$

1 $\xrightarrow{A} (s, [b]X)$

$\xrightarrow{A} (s_1, X)$ → jogo infinito como $X \stackrel{min}{=}$

~~2 $\xrightarrow{A} (s, \langle b\rangle tt)$~~

$\Downarrow$

~~2 $\xrightarrow{A} (s_1, \langle b\rangle tt)$~~

A ganha

HML

Ponto Fixo

Exercícios:

Maior e menor dos pontos fixos

$X \overset{max}{=} \overbrace{\langle b \rangle true \wedge [b] X}^{}$ ①   é sempre possível uma transição por $\underline{\underline{b}}$

$Y \overset{min}{=} \langle b \rangle true \vee \langle \{a, b\} \rangle Y \; (\langle a \rangle Y \vee \langle b \rangle Y)$ ②

① $[\![X]\!] \; Proc \quad [\![\overbrace{\langle b \rangle true \wedge [b] X}^{A}]\!]$

Nível 1:

$[\![A]\!] = \langle \cdot b \cdot \rangle Proc \cap [\cdot b \cdot] Proc$

$= \{s_1, s_2, t_1\} \cap \{s_1, s_2, t_1, s, t\}$

$= \{s_1, s_2, t_1\}$ (camada não é ponto fixo)

Nível 2:

$[\![X]\!] = \{s_1, s_2, t_1\}$

$[\![A]\!] = \langle \cdot b \cdot \rangle Proc \cap [\cdot b \cdot] \{s_1, s_2, t_1\}$

$= \{s_1, s_2, t_1\} \cap \{s_1, s_2, t_1, s, t\}$

$= \{s_1, s_2, t_1\}$

A fórmula é sempre válida para qq estado $s \in \{s_1, s_2, t_1\}$

② $[\![y]\!] = \{\}$

Nível 1:

$\beta = \langle b \rangle true \vee (\langle a \rangle y \vee \langle b \rangle y)$

$[\![\beta]\!] = \langle\cdot b\cdot\rangle PROC \cup (\langle\cdot a\cdot\rangle \{\} \cup \langle\cdot b\cdot\rangle \{\})$

$= \{s_1, s_2, t_1\} \cup \{\}$

$= \{s_1, s_2, t_1\}$ (ainda não é ponto fixo)

Nível 2:

$[\![y]\!] = \{s_1, s_2, t_1\}$

$[\![\beta]\!] = \langle\cdot b\cdot\rangle PROC \cup (\langle\cdot a\cdot\rangle \{s_1, s_2, t_1\} \cup$
$\langle\cdot b\cdot\rangle \{s_1, s_2, t_1\}$

$= \{s_1, s_2, t_1\} \cup \{s, t\} \cup \{s_1, s_2, t_1\}$

$= PROC$ (ainda não é ponto fixo)

Nível 3:

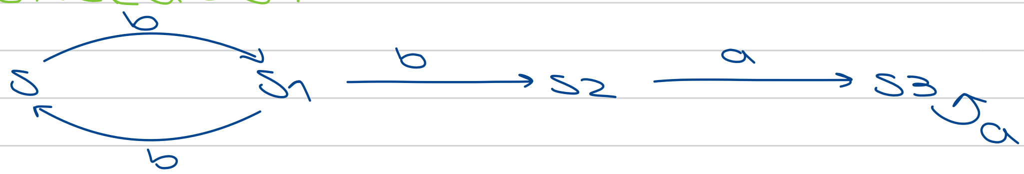
$[\![y]\!] = PROC$

$[\![\beta]\!] = \{s_1, s_2, t_1\} \cup \langle\cdot a\cdot\rangle PROC \cup \langle\cdot b\cdot\rangle PROC$

$= \{s_1, s_2, t_1\} \cup \{s, t\} \cup \{s_1, s_2, t_1\}$

$= PROC$

Exercício:



$s \models \langle a \rangle true \vee \langle b \rangle X$

1)

$[\![X]\!] = \{\}$

$[\![\langle a \rangle true \vee \langle b \rangle X]\!]$

$\langle\cdot a\cdot\rangle PROC \cup \langle\cdot b\cdot\rangle \{\}$

$= \{s_2, s_3\} \cup \{\} = \{s_2, s_3\}$

2)

$[\![X]\!] = \{s_2, s_3\}$

$\langle \cdot a \cdot \rangle \text{PROC} \cup \langle \cdot b \cdot \rangle \{s_2, s_3\}$

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

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

3)

$[\![X]\!] = \{s_2, s_3, s_1\}$

$\langle \cdot a \cdot \rangle \text{PROC} \cup \langle \cdot b \cdot \rangle \{s_2, s_3, s_1\}$

$= \{s_2, s_3\} \cup \{s_1, s\} = \text{PROC}$

4)

$[\![X]\!] = \text{PROC}$

$[\![\langle \cdot a \cdot \rangle \text{PROC} \cup \langle \cdot b \cdot \rangle \text{PROC}]\!]$

$= \{s_2, s_3\} \cup \{s, s_1\} = \text{PROC}$

Tendo:

$$X \overset{mm}{=} \langle a \rangle \text{true} \vee ([b] X \wedge \langle b \rangle \text{true})$$

verificar se $s_1 \overset{?}{\models}$ (é válido em s_1)

1)

$[\![X]\!] = \{\}$

$\langle \cdot a \cdot \rangle \text{PROC} \vee ([\cdot b] \{\} \cap \langle \cdot b \cdot \rangle \text{PROC})$

$= \{s_2, s_3\} \cup (\{s_2, s_3\} \cap \{s, s_1\})$

$= \{s_2, s_3\}$

2)

$[\![X]\!] = \{s_2, s_3\}$

$\{s_2, s_3\} \cup ([\cdot b \cdot] \{s_2, s_3\} \cap \{s, s_1\})$

$= \{s_2, s_3\} \cup (\{s_1, s_2, s_3\} \cap \{s, s_1\})$

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

3)

$[\![x]\!] = \{s_1, s_2, s_3\}$

$\{s_2, s_3\} \cup ([\cdot b \cdot] \{s_1, s_2, s_3\} \cap \{s, s_1\})$

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

4)

$[\![x]\!] = PROC$

$\langle \cdot a \cdot \rangle PROC \cup ([\cdot b \cdot] PROC \cap \langle \cdot b \cdot \rangle true)$

$= \{s_2, s_3\} \cup (\{s, s_1, s_2, s_3\} \cap \{s, s_1\})$

$= PROC$