LCD (17/03)

* Bisimilarity as a game

Bisimulation: relation $R \subseteq Proc \times Proc$ such that if $P \, R \, Q$

(i) if $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$.

(ii) if $Q \xrightarrow{\alpha} Q'$ then $P \xrightarrow{\alpha} P'$ and $P' \, R \, Q'$

→ P, Q bisimilar ($P \sim Q$) if there exists R bisimulation s.t. $P \, R \, Q$

i.e.

$$\sim \; = \; \bigcup \{ R \mid R \text{ is a bisimulation} \}$$

* how to show that $P \sim Q$?

→ take all possible relations R s.t. $P \, R \, Q$

→ check if R is a bisimulation

suppose P, Q have finitely many states ($n \in \mathbb{N}$ states)

$R \subseteq Proc \times Proc$ $\quad |Proc| = n$

$$|Proc \times Proc| = n^2$$

$$\# \text{relations} = 2^{n^2}$$

es: $n = 10$

$$2^{10^2} = 2^{100} \simeq 10^{30} \quad \text{(nonillion)}$$

* Bisimulation Game

for establishing if $P \sim Q$

2 players
- **A**TTACKER: aims at proving $P \not\sim Q$ by proposing challenges $P \xrightarrow{\alpha} P'$
- **D**EFENDER: aim at proving $P \sim Q$ by answering to challenges $Q \xrightarrow{\alpha} Q'$

more precisely :

- rounds : at each round the game is in a configuration (P', Q')

→ moves : Attacker chooses a side (left / right)
P' Q'

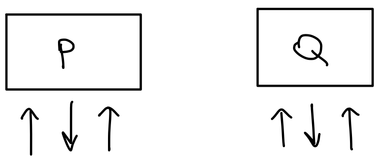
and a transition of the chosen process
$P' \xrightarrow{\alpha} P''$

Defender answers with a transition of the other process
$Q' \xrightarrow{\alpha} Q''$

and continue from (P'', Q'').

* PLAY : sequence of moves starting from (P, Q)
which is maximal
(1) finite, one of the players cannot move
(2) infinite

* who is the winner ?
(1) finite play : the player who played last
(2) infinite play : defender



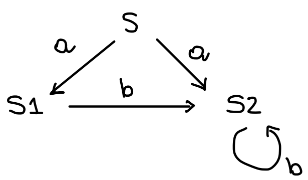
Theorem : Given P, Q processes [deepening]

- $P \sim Q$ iff the Defender has a winning strategy
- $P \not\sim Q$ iff the Attacker has a winning strategy

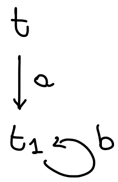
function Attacker : configuration (P', Q') ⇝ side (l/r) transition

Defender : configuration (P', Q') & attacker move
→ answer (move of other process)

Example :

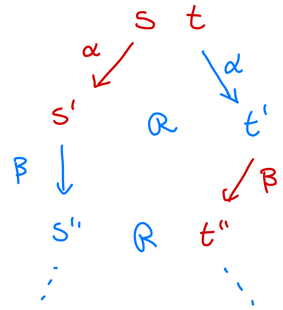


$\sim$



$$\mathcal{R} = \{ (s,t) , (s_1, t_1) , (s_2, t_1) \}$$

$$s \sim t$$



Example :

$$CTM = coin . (\overline{coffee} . CTM + \overline{tea} . CTM) \not\sim CTM' = coin . \overline{coffee} . CTM' + coin . \overline{tea} . CTM'$$

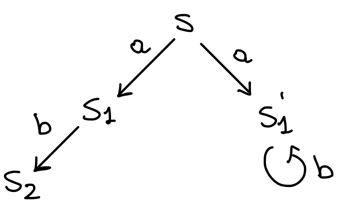
A : $CTM' \xrightarrow{coin} \overline{coffee} . CTM'$

D : $CTM \xrightarrow{coin} \overline{coffee} . CTM + \overline{tea} . CTM$

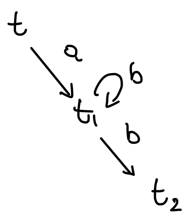
A : $\overline{coffee} . CTM + \overline{tea} . CTM \xrightarrow{\overline{tea}} CTM$

D : $\overline{coffee} . CTM'$ ~~$\xrightarrow{\overline{tea}}$~~

Example :



$\overset{?}{\not\sim}$



A : $s \xrightarrow{a} s_1$

D : $t \xrightarrow{a} t_1$

A : $t_1 \xrightarrow{b} t_1$

D : $s_1 \xrightarrow{b} s_2$

A : $t_1 \xrightarrow{b} t_1$

D : s_2 ~~$\xrightarrow{b}$~~

The defender can win a play (if the attacker is playing badly)

A: $s \xrightarrow{a} s_1$ D: $t \xrightarrow{a} t_1$

A: $s_1 \xrightarrow{b} s_2$ D: $t_1 \xrightarrow{b} t_2$

A looses.

* Properties of $\sim$

Bisimulation : relation $R \subseteq Proc \times Proc$ such that if $P \, R \, Q$

(i) if $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$.

(ii) if $Q \xrightarrow{\alpha} Q'$ then $P \xrightarrow{\alpha} P'$ and $P' \, R \, Q'$

→ P, Q bisimilar ($P \sim Q$) if there exists R bisimulation s.t $P \, R \, Q$

i.e.

$$\sim \; = \; \bigcup \{ R \mid R \text{ is a bisimulation} \}$$

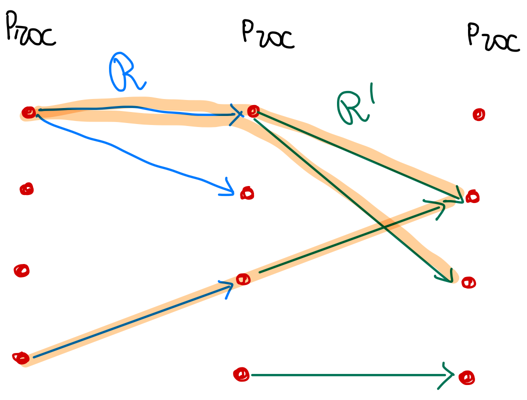
Is $\sim$ really an equivalence ?

OBSERVATION :

(i) $I = \{ (P, P) \mid P \in Proc \}$ bisimulation

(ii) if R is a bisimulation then $R^{-1} = \{ (Q, P) \mid (P, Q) \in R \}$ is a bisimulation

(iii) if R, R' are bisimulations then $R ; R' = \{ (P, S) \mid \exists Q \; P R Q \wedge Q R' S \}$ is bisimulation



(iv) if R_i $i \in I$ are bisimulations then $\bigcup_{i \in I} R_i$ is bisimulation

proof

(i) if $P \, I \, Q$ ($P = Q$) if $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ ($P = Q$, $P' = Q'$) and $P' \, I \, Q'$ ($P' = Q'$)

& dual

(ii) by symmetry of the definition

(iii) R, R' bisimulation

if $P\,(R;R')\,Q$

if $P \xrightarrow{\alpha} P'$ — we want to show → $Q \xrightarrow{\alpha} Q'$ and $P'\,(R;R')\,Q'$

there is S s.t. $P\,R\,S$ $S\,R'\,Q$

R bisim. → $S \xrightarrow{\alpha} S'$ and $P'\,R\,S'$

→ $Q \xrightarrow{\alpha} Q'$ and $S'\,R'\,Q'$

→ $P'\,(R;R')\,Q'$

& dual.

$\Rightarrow$ $R;R'$ is bisimulation.

(iv) R_i $i \in I$ bisimulation $\overset{?}{\Rightarrow}$ $\bigcup_{i\in I} R_i$ bisimulation

let $P\,(\bigcup_{i\in I} R_i)\,Q$

$\leadsto$ $\exists i \in I$ s.t. $P\,R_i\,Q$

if $P \xrightarrow{\alpha} P'$ → $Q \xrightarrow{\alpha} Q'$ and $P'\,R_i\,Q'$

hence $P'\,(\bigcup_{i\in I} R_i)\,Q'$

OBSERVATION: Bisimilarity is the largest bisimulation

i.e. (1) $\sim$ is a bisimulation ($\sim\, = \bigcup\{R \mid R \text{ bisimulation}\}$)

(2) $\sim$ is the largest bisimulation (i.e. R bisim $\Rightarrow R \subseteq \sim$)

proof

No need.

OBSERVATION : Bisimilarity $\sim$ is an equivalence

- reflexive

since I is a bisimulation : $\forall P \quad P \, I \, P \quad \leadsto \quad P \sim P$

- symmetric

if $P \sim Q$ then there is R bisim. s.t. $P R Q$ then $Q R^{-1} P$

↑ bisimulation

and thus $Q \sim P$

- transitive

if $P \sim Q$ and $Q \sim S$ then there are R, R' bisimulations st.

$P R Q \qquad Q R' S$

then $P \underbrace{R ; R'}_{\text{bisimulation}} S \quad \leadsto \quad P \sim S$.

□

EXERCISE : find a bisimulation which is not

- reflexive
- symmetric
- transitive

EXERCISE : Show

$P \sim Q \quad$ iff $\quad$ if $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ and $P' \sim Q'$

& dual

$\Rightarrow$

$\sim$ is a bisimulation

$\Leftarrow$
?

EXERCISE (E) : FINITE CCS

without constants

$$P, Q ::= \alpha.P \;\mid\; \sum_{i \in I} P_i \;\mid\; P \mid Q \;\mid\; P \setminus L \;\mid\; P[f]$$

I finite

properties

1. programs always terminate
2. " are finite state

(1) every computation

$$P \xrightarrow{\alpha_1} P_1 \xrightarrow{\alpha_2} \cdots \qquad \text{is finite}$$

proof idea:

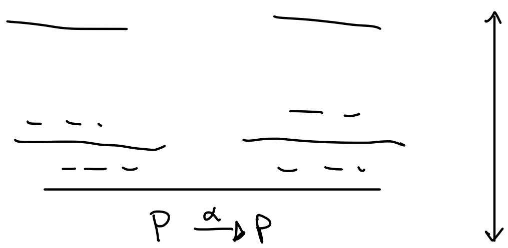
show that if $P \xrightarrow{\alpha_1} P_2 \xrightarrow{\alpha_2} \cdots \xrightarrow{\alpha_{m-1}} P_m$

then $m \leq |P|$ ↖ size of program

in turn this follows from

if $P \xrightarrow{\alpha} P'$ then $|P'| < |P|$

proof by induction on heigth of derivation (rule induction)



(2) every P has finite states

$$\text{States}(P) = \{ P' \mid P \xrightarrow{\alpha_1} P_1 \xrightarrow{\alpha_2} P_2 \cdots \xrightarrow{\alpha_m} P' \}$$

finite

define $\#(P)$ = bound for number of elements in states (P)

$$\# 0 = 1$$

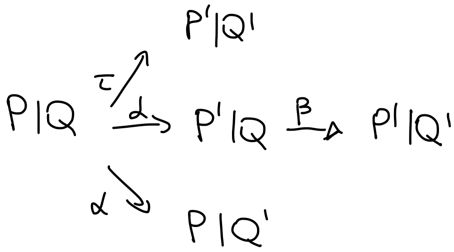
$$\# \alpha.P = 1 + \# P$$

$$\# \left(\sum_{i \in I} P_i\right) = \sum_{i \in I} \# P$$

$$\# (P | Q) = (\# P) * (\# Q)$$

$$\# (P \setminus L) = \# P$$

$$\# (P[f]) = \# P$$



prove that $|States(P)| \leq \# P$

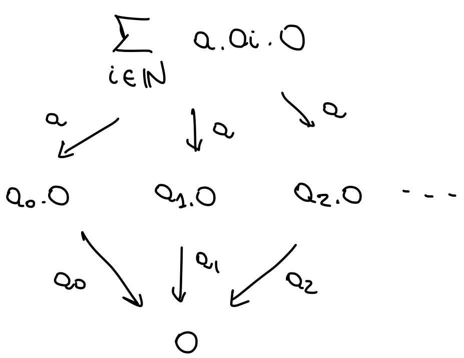
* what happens if we allow infinite non-deterministic choice ?

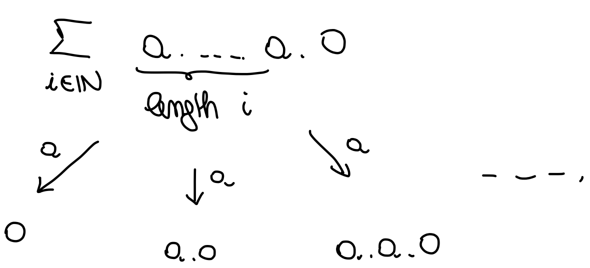
$$P, Q ::= \alpha.P \mid \sum_{i \in I} P_i \mid P | Q \mid P \setminus L \mid P[f]$$

~~I finite~~

(2) states can be infinitely many

$A = \{a\} \cup \{a_i \mid i \in \mathbb{N}\}$





(1) ?? is termination ensured ?