Bisimilarity and Behavioural Equivalences

Luís Soares Barbosa

HASLab - INESC TEC
Universidade do Minho
Braga, Portugal

February 2019

Behavioural Equivalences – Intuition

Two LTS should be equivalent if they cannot be distinguished by interacting with them.

Equality of functional behaviour

is not preserved by parallel composition: non compositional semantics, cf,

`x:=4; x:=x+1` and `x:=5`

Graph isomorphism

is too strong (why?)

Trace

Definition

Let $T = \langle S, N, \longrightarrow \rangle$ be a labelled transition system. The set of traces $\mathsf{Tr}(s)$, for $s \in S$ is the minimal set satisfying

$$(1) \quad \epsilon \in \mathsf{Tr}(s)$$

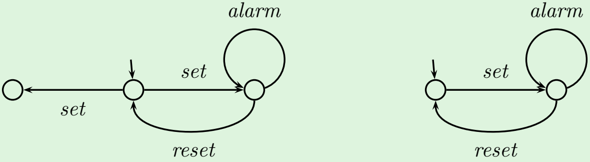
$$(2) \quad a\sigma \in \mathsf{Tr}(s) \;\Rightarrow\; \langle \exists\, s' \;:\; s' \in S : \; s \xrightarrow{a} s' \wedge \sigma \in \mathsf{Tr}(s') \rangle$$

Trace equivalence

Definition

Two states s, r are trace equivalent iff $\mathrm{Tr}(s) = \mathrm{Tr}(r)$
(i.e. if they can perform the same finite sequences of transitions)

Example



Trace equivalence applies when one can neither interact with a system, nor distinguish a slow system from one that has come to a stand still.

Simulation

the quest for a behavioural equality:
able to identify states that cannot be distinguished by any realistic form of observation

Simulation

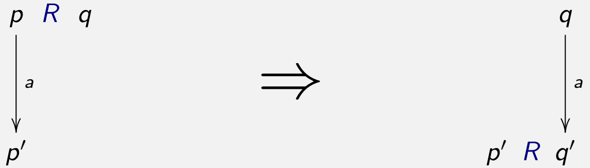
A state q simulates another state p if
every transition from q is corresponded by a transition from p and this capacity is kept along the whole life of the system to which state space q belongs to.

Simulation

Definition

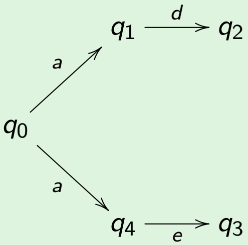
Given $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$ over N, relation $R \subseteq S_1 \times S_2$ is a simulation iff, for all $\langle p, q \rangle \in R$ and $a \in N$,

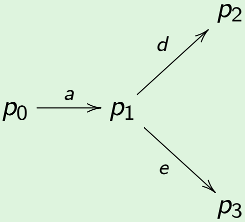
$$(1) \quad p \xrightarrow{a}_1 p' \;\Rightarrow\; \langle \exists\, q' \;:\; q' \in S_2 :\; q \xrightarrow{a}_2 q' \,\wedge\, \langle p', q' \rangle \in R \rangle$$



Example

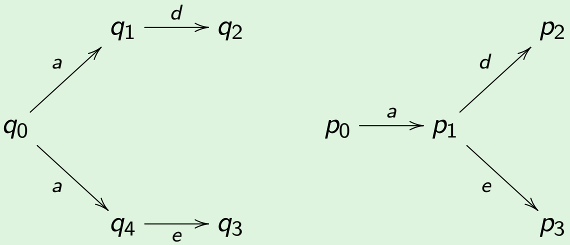
Find simulations





Example

Find simulations



$q_0 \lesssim p_0$ cf. $\{\langle q_0, p_0\rangle, \langle q_1, p_1\rangle, \langle q_4, p_1\rangle, \langle q_2, p_2\rangle, \langle q_3, p_3\rangle\}$

Similarity

Definition

$$p \lesssim q \;\equiv\; \langle \exists\, R \;::\; R \text{ is a simulation and } \langle p, q\rangle \in R\rangle$$

We say q simulates p.

Lemma

The similarity relation is a preorder
(ie, reflexive and transitive)

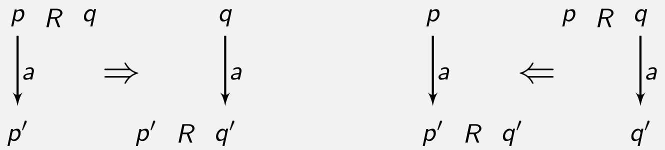
Bisimulation

Definition

Given $\langle S_1, N, \longrightarrow_1\rangle$ and $\langle S_2, N, \longrightarrow_2\rangle$ over N, relation $R \subseteq S_1 \times S_2$ is a **bisimulation** iff both R and its converse R° are simulations.
I.e., whenever $\langle p, q\rangle \in R$ and $a \in N$,

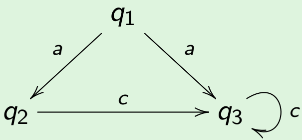
$$(1)\quad p \xrightarrow{a}_1 p' \;\Rightarrow\; \langle \exists\, q' \;:\; q' \in S_2 :\; q \xrightarrow{a}_2 q' \;\wedge\; \langle p', q'\rangle \in R\rangle$$

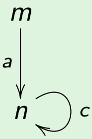
$$(2)\quad q \xrightarrow{a}_2 q' \;\Rightarrow\; \langle \exists\, p' \;:\; p' \in S_1 :\; p \xrightarrow{a}_1 p' \;\wedge\; \langle p', q'\rangle \in R\rangle$$

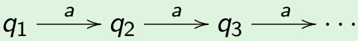


Examples

Find bisimulations

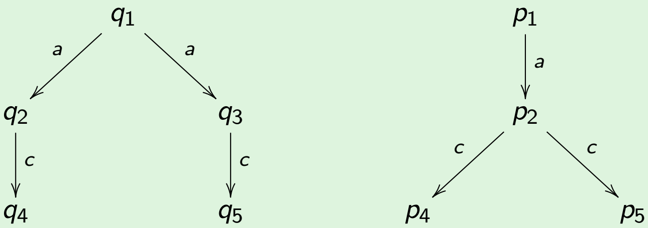


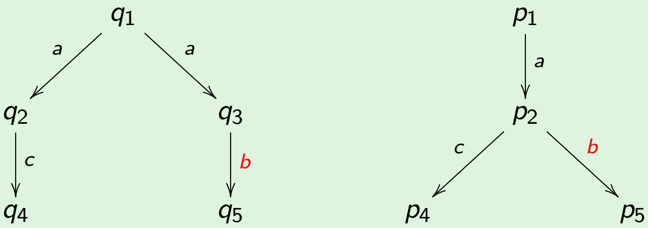




Examples

Find bisimulations





After thoughts

- Follows a $\forall, \exists$ pattern: p in all its transitions challenge q which is called to find a match to each of those (and conversely)
- Tighter correspondence with transitions
- Based on the information that the transitions convey, rather than on the shape of the LTS
- Local checks on states
- Lack of hierarchy on the pairs of the bisimulation (no temporal order on the checks is required)

which means bisimilarity can be used to reason about infinite or circular behaviours.

After thoughts

Compare the definition of bisimilarity with

$p == q$ if, for all $a \in N$

$$(1)\quad p \xrightarrow{a}_1 p' \;\Rightarrow\; \langle \exists\, q' \;:\; q' \in S_2 :\; q \xrightarrow{a}_2 q' \wedge p' == q' \rangle$$

$$(2)\quad q \xrightarrow{a}_2 q' \;\Rightarrow\; \langle \exists\, p' \;:\; p' \in S_1 :\; p \xrightarrow{a}_1 p' \wedge p' == q' \rangle$$

After thoughts

$p == q$ if, for all $a \in N$

$$(1)\ \ p \xrightarrow{a}_1 p' \ \Rightarrow\ \langle \exists\ q' \ :\ q' \in S_2 :\ q \xrightarrow{a}_2 q' \wedge p' == q' \rangle$$

$$(2)\ \ q \xrightarrow{a}_2 q' \ \Rightarrow\ \langle \exists\ p' \ :\ p' \in S_1 :\ p \xrightarrow{a}_1 p' \wedge p' == q' \rangle$$

- The meaning of $==$ on the pair $\langle p, q \rangle$ requires having already established the meaning of $==$ on the derivatives
- ... therefore the definition is ill-founded if the state space reachable from $\langle p, q \rangle$ is infinite or contain loops
- ... this is a local but inherently inductive definition (to revisit later)

After thoughts

Proof method

To prove that two behaviours are bisimilar, find a bisimulation containing them ...

- ... impredicative character
- coinductive vs inductive definition

Properties

Definition

$$p \sim q \;\equiv\; \langle \exists\, R \;::\; R \text{ is a bisimulation and } \langle p, q\rangle \in R\rangle$$

Lemma

1. The identity relation id is a bisimulation
2. The empty relation $\perp$ is a bisimulation
3. The converse R° of a bisimulation is a bisimulation
4. The composition $S \cdot R$ of two bisimulations S and R is a bisimulation
5. The $\bigcup_{i \in I} R_i$ of a family of bisimulations $\{R_i \mid i \in I\}$ is a bisimulation

Properties

Lemma

The bisimilarity relation is an equivalence relation
(ie, reflexive, symmetric and transitive)

Lemma

The class of all bisimulations between two LTS has the structure of a complete lattice, ordered by set inclusion, whose top is the bisimilarity relation $\sim$.

Properties

Lemma

In a deterministic labelled transition system, two states are bisimilar iff they are trace equivalent, i.e.,

$$s \sim s' \;\Leftrightarrow\; \mathrm{Tr}(s) = \mathrm{Tr}(s')$$

Hint: define a relation R as

$$\langle x, y \rangle \in R \;\Leftrightarrow\; \mathrm{Tr}(x) = \mathrm{Tr}(y)$$

and show R is a bisimulation.

Properties

Warning

The bisimilarity relation $\sim$ is not the symmetric closure of $\lesssim$

i.e., $\left[p \lesssim q \text{ and } q \lesssim p\right]$ does not imply $\left[p \sim q\right]$

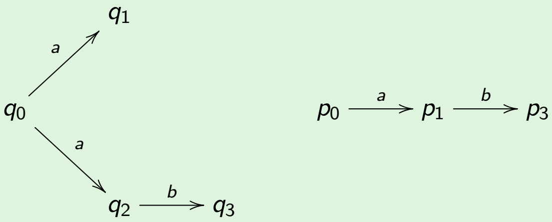
Properties

Warning

The bisimilarity relation $\sim$ is not the symmetric closure of $\lesssim$

Example

$$q_0 \lesssim p_0,\ p_0 \lesssim q_0 \quad \text{but} \quad p_0 \not\sim q_0$$



Notes

Similarity as the greatest simulation

$$\lesssim \triangleq \bigcup\{S \mid S \text{ is a simulation}\}$$

Bisimilarity as the greatest bisimulation

$$\sim \triangleq \bigcup\{S \mid S \text{ is a bisimulation}\}$$

Exercises

P,Q Bisimilar?

$$P = a.P_1$$
$$P_1 = b.P + c.P$$

$$Q = a.Q_1$$
$$Q_1 = b.Q_2 + c.Q$$
$$Q_2 = a.Q_3$$
$$Q_3 = b.Q + c.Q_2$$

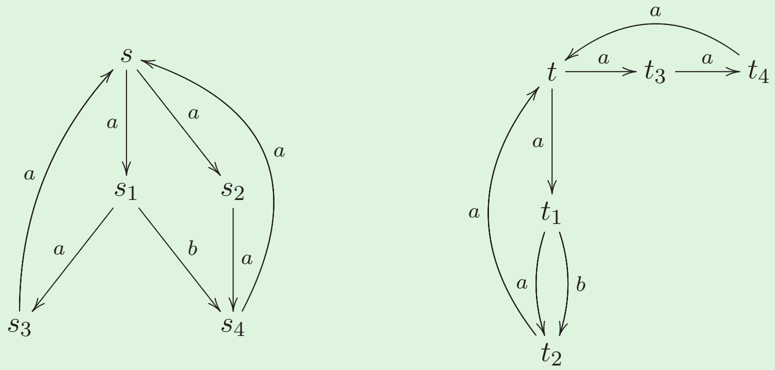
P,Q Bisimilar?

$$P = a.(b.\mathbf{0} + c.\mathbf{0})$$

$$Q = a.b.\mathbf{0} + a.c.\mathbf{0}$$

Exercises

Find a bisimulation



Processes are 'prototipycal' transition systems

Example: $S \sim M$

$$T \triangleq i.\overline{k}.T$$
$$R \triangleq k.j.R$$
$$S \triangleq (T \mid R)\backslash\{k\}$$

$$M \triangleq i.\tau.N$$
$$N \triangleq j.i.\tau.N + i.j.\tau.N$$

through bisimulation

$$R = \{\langle S, M\rangle, \langle(\overline{k}.T \mid R)\backslash\{k\}, \tau.N\rangle, \langle(T \mid j.R)\backslash\{k\}, N\rangle, \langle(\overline{k}.T \mid j.R)\backslash\{k\}, j.\tau.N\rangle\}$$

Example: Semaphores

A semaphore

$$Sem \triangleq get.put.Sem$$

n-semaphores

$$\begin{aligned}
Sem_n &\triangleq Sem_{n,0} \\
Sem_{n,0} &\triangleq get.Sem_{n,1} \\
Sem_{n,i} &\triangleq get.Sem_{n,i+1} + put.Sem_{n,i-1} \\
&\quad (\text{for } 0 < i < n) \\
Sem_{n,n} &\triangleq put.Sem_{n,n-1}
\end{aligned}$$

Sem_n can also be implemented by the parallel composition of n Sem processes:

$$Sem^n \triangleq Sem \mid Sem \mid \ldots \mid Sem$$

Example: Semaphores

A semaphore

$$Sem \triangleq get.put.Sem$$

n-semaphores

$$\begin{aligned}
Sem_n &\triangleq Sem_{n,0} \\
Sem_{n,0} &\triangleq get.Sem_{n,1} \\
Sem_{n,i} &\triangleq get.Sem_{n,i+1} + put.Sem_{n,i-1} \\
&\quad (\text{for } 0 < i < n) \\
Sem_{n,n} &\triangleq put.Sem_{n,n-1}
\end{aligned}$$

Sem_n can also be implemented by the parallel composition of n Sem processes:

$$Sem^n \triangleq Sem \mid Sem \mid \dots \mid Sem$$

Example: Semaphores

Is $Sem_n \sim Sem^n$?

For $n = 2$:

$$\{\langle Sem_{2,0}, Sem \mid Sem\rangle, \langle Sem_{2,1}, Sem \mid put.Sem\rangle,$$
$$\langle Sem_{2,1}, put.Sem \mid Sem\rangle\langle Sem_{2,2}, put.Sem \mid put.Sem\rangle\}$$

is a bisimulation.

- but can we get rid of structurally congruent pairs?

Example: Semaphores

Is $Sem_n \sim Sem^n$?

For $n = 2$:

$$\{\langle Sem_{2,0}, Sem \mid Sem\rangle, \langle Sem_{2,1}, Sem \mid put.Sem\rangle,$$
$$\langle Sem_{2,1}, put.Sem \mid Sem\rangle\langle Sem_{2,2}, put.Sem \mid put.Sem\rangle\}$$

is a bisimulation.

- but can we get rid of structurally congruent pairs?

Semantics

Structural congruence

$\equiv$ over $\mathbb{P}$ is given by the closure of the following conditions:

- for all $A(\tilde{x}) \triangleq E_A$, $A(\tilde{y}) \equiv \{\tilde{y}/\tilde{x}\}\, E_A$,
  (*i.e.*, folding/unfolding preserve $\equiv$)
- α-conversion (*i.e.*, replacement of bounded variables).
- both $|$ and $+$ originate, with $\mathbf{0}$, Abelian monoids
- forall $a \notin \mathrm{fn}(P)$ $(P \mid Q)\backslash\{a\} \equiv P \mid Q\backslash\{a\}$
- $\mathbf{0}\backslash\{a\} \equiv \mathbf{0}$

Bisimulation up to $\equiv$

Definition

A binary relation S in $\mathbb{P}$ is a (strict) bisimulation up to $\equiv$ iff, whenever $(E,F) \in S$ and $a \in Act$,

$$\text{i) } E \xrightarrow{a} E' \Rightarrow F \xrightarrow{a} F' \wedge (E',F') \in \equiv \cdot S \cdot \equiv$$

$$\text{ii) } F \xrightarrow{a} F' \Rightarrow E \xrightarrow{a} E' \wedge (E',F') \in \equiv \cdot S \cdot \equiv$$

Lemma

If S is a (strict) bisimulation up to $\equiv$, then $S \subseteq \sim$

- To prove $Sem_n \sim Sem^n$ a bisimulation will contain 2^n pairs, while a bisimulation up to $\equiv$ only requires $n+1$ pairs.

Bisimulation up to $\equiv$

Definition

A binary relation S in $\mathbb{P}$ is a (strict) bisimulation up to $\equiv$ iff, whenever $(E, F) \in S$ and $a \in Act$,

$$\text{i) } E \xrightarrow{a} E' \;\Rightarrow\; F \xrightarrow{a} F' \;\wedge\; (E', F') \in \equiv \cdot S \cdot \equiv$$

$$\text{ii) } F \xrightarrow{a} F' \;\Rightarrow\; E \xrightarrow{a} E' \;\wedge\; (E', F') \in \equiv \cdot S \cdot \equiv$$

Lemma

If S is a (strict) bisimulation up to $\equiv$, then $S \subseteq \sim$

- To prove $Sem_n \sim Sem^n$ a bisimulation will contain 2^n pairs, while a bisimulation up to $\equiv$ only requires $n+1$ pairs.

Bisimulation up to $\equiv$

Definition

A binary relation S in $\mathbb{P}$ is a (strict) bisimulation up to $\equiv$ iff, whenever $(E, F) \in S$ and $a \in Act$,

$$\text{i)}\ E \xrightarrow{a} E' \ \Rightarrow\ F \xrightarrow{a} F' \wedge (E', F') \in \equiv \cdot S \cdot \equiv$$

$$\text{ii)}\ F \xrightarrow{a} F' \ \Rightarrow\ E \xrightarrow{a} E' \wedge (E', F') \in \equiv \cdot S \cdot \equiv$$

Lemma

If S is a (strict) bisimulation up to $\equiv$, then $S \subseteq \sim$

- To prove $Sem_n \sim Sem^n$ a bisimulation will contain 2^n pairs, while a bisimulation up to $\equiv$ only requires $n + 1$ pairs.

A $\sim$-calculus

Lemma $E \equiv F \;\Rightarrow\; E \sim F$

- proof idea: show that $\{(E+E, E) \mid E \in \mathbb{P}\} \cup Id_{\mathbb{P}}$ is a bisimulation

Lemma

$$(E \backslash K) \backslash K' \sim E \backslash (K \cup K')$$

$$E \backslash K \sim E \qquad \text{if } \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset$$

$$(E \mid F) \backslash K \sim E \backslash K \mid F \backslash K \qquad \text{if } \mathbb{L}(E) \cap \overline{\mathbb{L}(F)} \cap (K \cup \overline{K}) = \emptyset$$

- proof idea: discuss whether S is a bisimulation:

$$S \;=\; \{(E \backslash K, E) \mid E \in \mathbb{P} \wedge \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset\}$$

A $\sim$-calculus

Lemma $E \equiv F \;\Rightarrow\; E \sim F$

- proof idea: show that $\{(E + E, E) \mid E \in \mathbb{P}\} \cup Id_{\mathbb{P}}$ is a bisimulation

Lemma

$$(E \backslash K) \backslash K' \sim E \backslash (K \cup K')$$

$$E \backslash K \sim E \qquad \text{if } \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset$$

$$(E \mid F) \backslash K \sim E \backslash K \mid F \backslash K \qquad \text{if } \mathbb{L}(E) \cap \overline{\mathbb{L}(F)} \cap (K \cup \overline{K}) = \emptyset$$

- proof idea: discuss whether S is a bisimulation:

$$S \;=\; \{(E \backslash K, E) \mid E \in \mathbb{P} \wedge \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset\}$$

$\sim$ is a congruence

congruence is the name of modularity in Mathematics

- process combinators preserve $\sim$

Lemma

Assume $E \sim F$. Then,

$$a.E \sim a.F$$
$$E + P \sim F + P$$
$$E \mid P \sim F \mid P$$
$$E \backslash K \sim F \backslash K$$

- recursive definition preserves $\sim$

$\sim$ is a congruence

congruence is the name of modularity in Mathematics

- process combinators preserve $\sim$

Lemma

Assume $E \sim F$. Then,

$$a.E \sim a.F$$
$$E + P \sim F + P$$
$$E \mid P \sim F \mid P$$
$$E\backslash K \sim F\backslash K$$

- recursive definition preserves $\sim$

The expansion theorem

Every process is equivalent to the sum of its derivatives

$$E \sim \sum \{a.E' \mid E \xrightarrow{a} E'\}$$

Example

$S \sim M$

$$
\begin{aligned}
S &\sim (T \mid R)\backslash\{k\} \\
&\sim i.(\overline{k}.T \mid R)\backslash\{k\} \\
&\sim i.\tau.(T \mid j.R)\backslash\{k\} \\
&\sim i.\tau.(i.(\overline{k}.T \mid j.R)\backslash\{k\} + j.(T \mid R)\backslash\{k\}) \\
&\sim i.\tau.(i.j.(\overline{k}.T \mid R)\backslash\{k\} + j.i.(\overline{k}.T \mid R)\backslash\{k\}) \\
&\sim i.\tau.(i.j.\tau.(T \mid j.R)\backslash\{k\} + j.i.\tau.(T \mid j.R)\backslash\{k\})
\end{aligned}
$$

Let $N' = (T \mid j.R)\backslash\{k\}$.
This expands into $N' \sim i.j.\tau.(T \mid j.R)\backslash\{k\} + j.i.\tau.(T \mid j.R)\backslash\{k\}$,
Therefore $N' \sim N$ and $S \sim i.\tau.N \sim M$

- requires result on unique solutions for recursive process equations

Observable transitions

$$\stackrel{a}{\Longrightarrow} \subseteq \mathbb{P} \times \mathbb{P}$$

- $L \cup \{\epsilon\}$
- A $\stackrel{\epsilon}{\Longrightarrow}$-transition corresponds to zero or more non observable transitions
- inference rules for $\stackrel{a}{\Longrightarrow}$:

$$\frac{}{E \stackrel{\epsilon}{\Longrightarrow} E} \, (O_1)$$

$$\frac{E \stackrel{\tau}{\longrightarrow} E' \quad E' \stackrel{\epsilon}{\Longrightarrow} F}{E \stackrel{\epsilon}{\Longrightarrow} F} \, (O_2)$$

$$\frac{E \stackrel{\epsilon}{\Longrightarrow} E' \quad E' \stackrel{a}{\longrightarrow} F' \quad F' \stackrel{\epsilon}{\Longrightarrow} F}{E \stackrel{a}{\Longrightarrow} F} \, (O_3) \quad \text{for } a \in L$$

Example

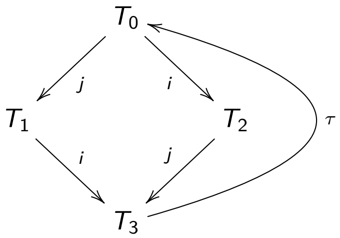
$$T_0 \triangleq j.T_1 + i.T_2$$
$$T_1 \triangleq i.T_3$$
$$T_2 \triangleq j.T_3$$
$$T_3 \triangleq \tau.T_0$$

and

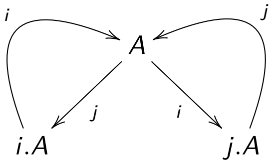
$$A \triangleq i.j.A + j.i.A$$

Example

From their graphs,



and



we conclude that $T_0 \approx A$ (why?).

Observational equivalence

$E \approx F$

- Processes E, F are observationally equivalent if there exists a weak bisimulation S st $\{\langle E, F\rangle\} \in S$.
- A binary relation S in $\mathbb{P}$ is a weak bisimulation iff, whenever $(E, F) \in S$ and $a \in L \cup \{\epsilon\}$,

$$\begin{array}{ll} \text{i)} & E \stackrel{a}{\Longrightarrow} E' \;\Rightarrow\; F \stackrel{a}{\Longrightarrow} F' \;\wedge\; (E', F') \in S \\ \text{ii)} & F \stackrel{a}{\Longrightarrow} F' \;\Rightarrow\; E \stackrel{a}{\Longrightarrow} E' \;\wedge\; (E', F') \in S \end{array}$$

I.e.,

$$\approx \;=\; \bigcup\{S \subseteq \mathbb{P} \times \mathbb{P} \mid S \text{ is a weak bisimulation}\}$$

Observational equivalence

Properties

- as expected: $\approx$ is an equivalence relation
- basic property: for any $E \in \mathbb{P}$,

$$E \;\approx\; \tau.E$$

(proof idea: $id_{\mathbb{P}} \cup \{(E, \tau.E) \mid E \in \mathbb{P}\}$ is a weak bisimulation

- weak vs. strict:

$$\sim \subseteq \approx$$

Is $\approx$ a congruence?

Lemma

Let $E \approx F$. Then, for any $P \in \mathbb{P}$ and $K \subseteq L$,

$$a.E \approx a.F$$
$$E \mid P \approx F \mid P$$
$$E \backslash K \approx F \backslash K$$

but

$$E + P \approx F + P$$

does not hold, in general.

Is $\approx$ a congruence?

Lemma

Let $E \approx F$. Then, for any $P \in \mathbb{P}$ and $K \subseteq L$,

$$a.E \approx a.F$$
$$E \mid P \approx F \mid P$$
$$E \backslash K \approx F \backslash K$$

but

$$E + P \approx F + P$$

does not hold, in general.

Is $\approx$ a congruence?

Example (initial τ restricts options 'menu')

$$i.\mathbf{0} \approx \tau.i.\mathbf{0}$$

However

$$j.\mathbf{0} + i.\mathbf{0} \not\approx j.\mathbf{0} + \tau.i.\mathbf{0}$$

Actually,

Is $\approx$ a congruence?

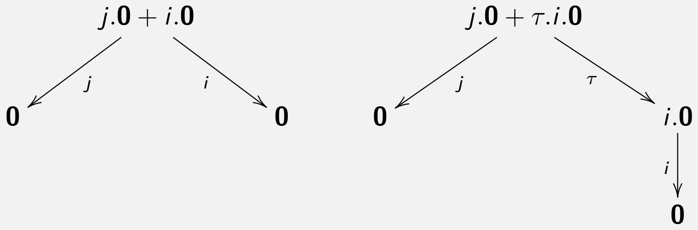
Example (initial τ restricts options 'menu')

$$i.\mathbf{0} \;\approx\; \tau.i.\mathbf{0}$$

However

$$j.\mathbf{0} + i.\mathbf{0} \not\approx j.\mathbf{0} + \tau.i.\mathbf{0}$$

Actually,



Forcing a congruence: $E = F$

Solution: force any initial τ to be matched by another τ

Process equality

Two processes E and F are equal (or observationally congruent) iff

i) $E \approx F$

ii) $E \xrightarrow{\tau} E' \;\Rightarrow\; F \xrightarrow{\tau} X \xRightarrow{\epsilon} F'$ and $E' \approx F'$

iii) $F \xrightarrow{\tau} F' \;\Rightarrow\; E \xrightarrow{\tau} X \xRightarrow{\epsilon} E'$ and $E' \approx F'$

- note that $E \neq \tau.E$, but $\tau.E = \tau.\tau.E$

Forcing a congruence: $E = F$

Solution: force any initial τ to be matched by another τ

Process equality

Two processes E and F are equal (or observationally congruent) iff

i) $E \approx F$

ii) $E \xrightarrow{\tau} E' \;\Rightarrow\; F \xrightarrow{\tau} X \stackrel{\epsilon}{\Longrightarrow} F'$ and $E' \approx F'$

iii) $F \xrightarrow{\tau} F' \;\Rightarrow\; E \xrightarrow{\tau} X \stackrel{\epsilon}{\Longrightarrow} E'$ and $E' \approx F'$

- note that $E \neq \tau.E$, but $\tau.E = \tau.\tau.E$

Forcing a congruence: $E = F$

$=$ can be regarded as a restriction of $\approx$ to all pairs of processes which preserve it in additive contexts

Lemma

Let E and F be processes st the union of their sorts is distinct of L. Then,

$$E = F \;\equiv\; \forall_{G \in \mathbb{P}} .\, (E + G \approx F + G)$$

Properties of $=$

Lemma

$$E \approx F \;\equiv\; (E = F) \vee (E = \tau.F) \vee (\tau.E = F)$$

- note that $E \neq \tau.E$, but $\tau.E = \tau.\tau.E$

Properties of $=$

Lemma

$$\sim \; \subseteq \; = \; \subseteq \; \approx$$

So,

the whole $\sim$ theory remains valid

Additionally,

Lemma (additional laws)

$$a.\tau.E = a.E$$
$$E + \tau.E = \tau.E$$
$$a.(E + \tau.F) = a.(E + \tau.F) + a.F$$