Communication and Concurrency

Insup Lee
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104

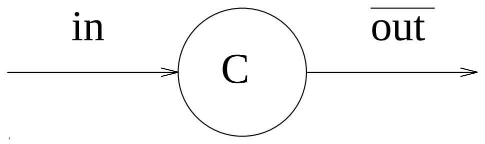
May 20, 1992

1 Introduction

Define the behavior of a system to be nothing more or less than its entire capability of communication

The behavior of a system is exactly what is observable, and to observe a system is exactly to communicate with it.

EX A cell which may hold a single data item.



$$C \stackrel{\text{def}}{=} in(x).C'(x)$$

$$C'(x) \stackrel{\text{def}}{=} \overline{out}(x).C$$

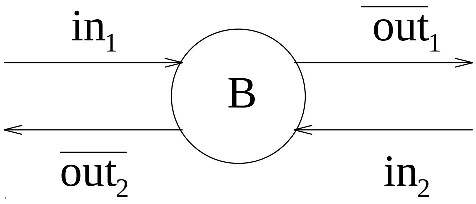
or

$$C \stackrel{\text{def}}{=} in(x).\overline{out}(x).C$$

EX A cell which may hold two data items.

$$A \stackrel{\text{def}}{=} in(x).in(y).\overline{out}(x).\overline{out}(y).A$$

EX A two-way buffer.



$$B \stackrel{\text{def}}{=} in_1(x).\overline{out_1}(x).B + in_2(x).\overline{out_2}(x).B$$

$P + Q$ behaves like either P or Q.

(External choice $+$: if both offered by environment then the choice is nondeterministic.)

Basic Definitions

$\mathcal{A}$: an infinite set of names $(a, b, \ldots)$

$\bar{\mathcal{A}}$: the infinite set of co-names $(\bar{a}, \bar{b}, \ldots)$

$\mathcal{L} = \mathcal{A} \cup \bar{\mathcal{A}}$: the set of labels $(l, l', \ldots)$

Let $\bar{\bar{a}} = a$.

Transition

$$P \xrightarrow{l} Q$$

EX $in.\overline{out}.C \xrightarrow{in} \overline{out}.C$

Composition $P|Q$

EX

$$A \stackrel{\text{def}}{=} a.A' \qquad B \stackrel{\text{def}}{=} c.B'$$
$$A' \stackrel{\text{def}}{=} \bar{c}.A \qquad B' \stackrel{\text{def}}{=} \bar{b}.B$$

Since $A \xrightarrow{a} A'$, we infer $A|B \xrightarrow{a} A'|B$.

Since $A' \xrightarrow{\bar{c}} A$, we infer $A'|B \xrightarrow{\bar{c}} A|B$.

To represent all handshake communications (any pair $(c, \bar{c})$),

a single perfect action τ is introduced.

$Act = \mathcal{L} \cup \{\tau\}$: the set of all possible actions $(\alpha, \beta, \ldots)$

Since $A' \xrightarrow{\bar{c}} A$ and $B \xrightarrow{\bar{c}} B'$, we infer $A'|B \xrightarrow{\tau} A|B'$.

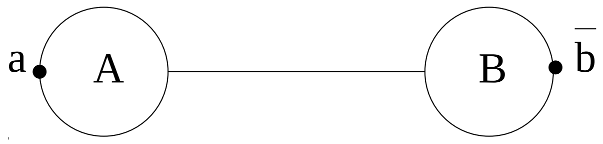
In general,

$P \xrightarrow{\tau} P_1 \xrightarrow{\tau} \cdots \xrightarrow{\tau} P_n$ is (observational) equivalent to $P \xrightarrow{\tau} P_n$.

Restriction $P\backslash L$

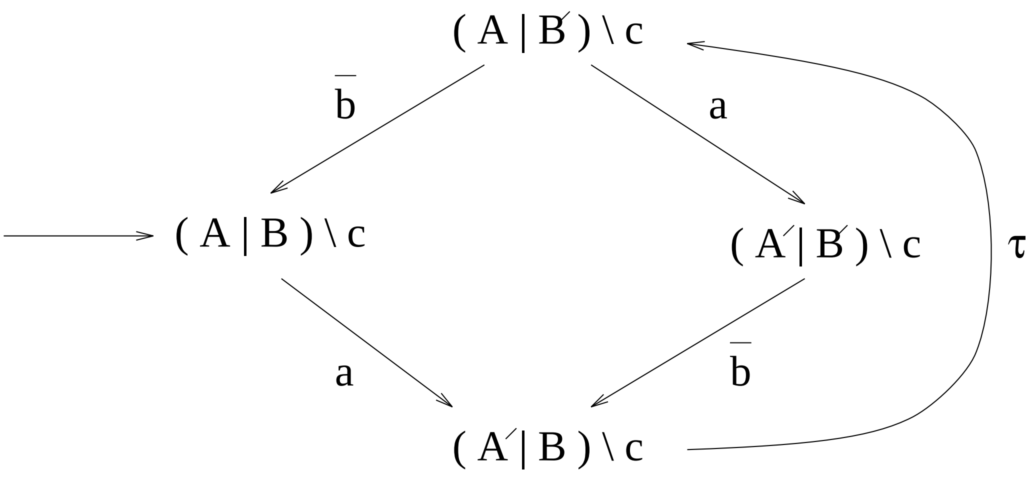
If $P \overset{\alpha}{\to} P'$, then $P\backslash L \overset{\alpha}{\to} P'\backslash L$ provided $\alpha, \bar{\alpha} \notin L$.

EX $(A|B)\backslash c$



$(A|B)\backslash c$ may not perform c or $\bar{c}$ action.

$(A|B)\backslash c$ may perform τ action which resulted from c and $\bar{c}$.



$(A|B)\backslash c$ behaviorally equal to C_1, where

$$
\begin{aligned}
C_0 &\overset{\text{def}}{=} \bar{b}.C_1 + a.C_2 \\
C_1 &\overset{\text{def}}{=} a.C_3 \\
C_2 &\overset{\text{def}}{=} \bar{b}.C_3 \\
C_3 &\overset{\text{def}}{=} \tau.C_0
\end{aligned}
$$

Alternatively,

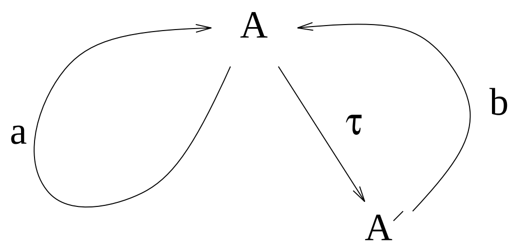
$$(A|B)\backslash c = a.\tau.C \text{ where } C \stackrel{\text{def}}{=} a.\bar{b}.\tau.C + \bar{b}.a.\tau.C$$

In general, $\alpha.\tau.P = \alpha.P$.

Thus, $(A|B)\backslash c = a.D$ where $D \stackrel{\text{def}}{=} a.\bar{b}.D + \bar{b}.a.D$.

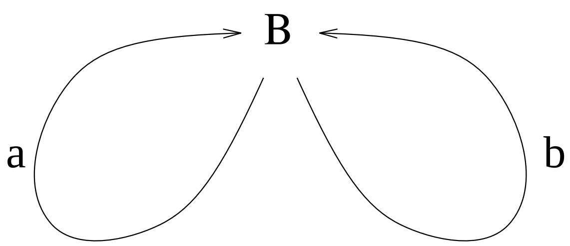
Note τ cannot always be dropped.

$$A \stackrel{\text{def}}{=} a.A + \tau.b.A$$



If we were allowed to drop τ, then $A = B$ where

$$B \stackrel{\text{def}}{=} a.B + b.B$$



B : a and b are always possible.

A : at A', only b is possible.

Comparison : In classical automata theory, two machines are normally taken to be equivalent iff they can perform the same action sequence.

The Basic Language

$\mathcal{A}$: names $(a, b, c, \ldots)$

$\bar{\mathcal{A}}$: co-names $(\bar{a}, \bar{b}, \bar{c}, \ldots)$

$\mathcal{L} = \mathcal{A} \cup \bar{\mathcal{A}}$: labels $(l, l', \ldots)$

$Act = \mathcal{L} \cup \{\tau\}$: actions $(\alpha, \beta, \ldots)$

$f : \mathcal{L} \to \mathcal{L}$ relabelling function such that $f(\bar{l}) = \overline{f(l)}$

f is extended to Act by $f(\tau) = \tau$.

$\mathcal{X}$: agent variables $(X, Y, \ldots)$

$\mathcal{K}$: agent constants $(A, B, \ldots)$

I, J : indexing sets

$\{E_i | i \in I\}$: a family of expressions indexed by I.

$\mathcal{E}$: the set of agent expressions $(E, F, \ldots)$

is the smallest set which includes $\mathcal{X}$ and $\mathcal{K}$,

and contains the following expressions, where $E, E_i \in \mathcal{E}$:

(1) $\alpha.E$, a *Prefix* $(\alpha \in Act)$

(2) $\Sigma_{i \in I} E_i$, a *Summation* (I an indexing set)

$E_1 + E_2 = \Sigma_{i \in I} E_i$ where $I = \{1, 2\}$

(3) $E_1 | E_2$, a *Composition*

(4) $E \backslash L$, a *Restriction* $(L \subseteq \mathcal{L})$

(5) $E[f]$, a *Relabelling* (f a relabelling function)

An agent expression E is an *agent* if it contains no free variables

$\mathcal{P}$: the set of agents $(P, Q, \ldots)$

$\mathbf{0}$: the inactive agent $(\mathbf{0} \stackrel{\mathrm{def}}{=} \Sigma_{i \in \emptyset} E_i)$

A : a constant agent $(A \stackrel{\mathrm{def}}{=} P)$

$E_1 \equiv E_2$: E_1 and E_2 are syntactically identical.

$E_1 = E_2$: an equivalence relation

A Labelled Transition System

$$(S, T, \{\xrightarrow{t} : t \in T\})$$

where S : a set of *states*,

T : a set of *transition labels*,

$\xrightarrow{t} \subseteq S \times S$: a *transition relation* for each $t \in T$.

Let S be $\mathcal{E}$ and T be Act.

Transition Rules

$$\textbf{Act}\frac{}{\alpha.E \xrightarrow{\alpha} E} \qquad \textbf{Sum}_\mathbf{j}\frac{E_j \xrightarrow{\alpha} E_j'}{\Sigma_{i\in I} E_i \xrightarrow{\alpha} E_j'}(j \in I)$$

$$\textbf{Com}_\mathbf{1}\frac{E \xrightarrow{\alpha} E'}{E|F \xrightarrow{\alpha} E'|F} \qquad \textbf{Com}_\mathbf{2}\frac{F \xrightarrow{\alpha} F'}{E|F \xrightarrow{\alpha} E|F'}$$

$$\textbf{Com}_\mathbf{3}\frac{E \xrightarrow{l} E',\ F \xrightarrow{\bar{l}} F'}{E|F \xrightarrow{\tau} E'|F'}$$

$$\textbf{Res}\frac{E \xrightarrow{\alpha} E'}{E\backslash L \xrightarrow{\alpha} E'\backslash L}(\alpha, \bar{\alpha} \notin L) \qquad \textbf{Rel}\frac{E \xrightarrow{\alpha} E'}{E[f] \xrightarrow{f(\alpha)} E'[f]}$$

$$\textbf{Con}\frac{P \xrightarrow{\alpha} P'}{A \xrightarrow{\alpha} P'}(A \stackrel{\text{def}}{=} P)$$

EX The justification of the transition

$$((a.E + b.\mathbf{0})|\bar{a}.F)\backslash a \xrightarrow{\tau} (E|F)\backslash a$$

$$\textbf{Act}\ \frac{}{a.E \xrightarrow{a} E}$$

$$\textbf{Sum}_\mathbf{1}\ \frac{}{a.E + b.\mathbf{0} \xrightarrow{a} E} \qquad \textbf{Act}\ \frac{}{\bar{a}.F \xrightarrow{\bar{a}} F}$$

$$\textbf{Com}_\mathbf{3}\ \frac{}{((a.E + b.\mathbf{0})|\bar{a}.F \xrightarrow{\tau} E|F}$$

$$\textbf{Res}\ \frac{}{((a.E + b.\mathbf{0})|\bar{a}.F\backslash a \xrightarrow{\tau} (E|F)\backslash a}$$

$E \xrightarrow{\alpha} E'$

α : an *action* of E

E' : an α-*derivative* of E

$E \xrightarrow{\alpha_1} \cdots \xrightarrow{\alpha_n} E'$

$\alpha_1 \cdots \alpha_n$: an *action-sequence* of E

E' : an $\alpha_1 \cdots \alpha_n$-*derivative* of E

In the case $n = 0$,

ϵ : an action sequence of E

E : a derivative of itself

Value-Passing Caculus

$$\begin{aligned} C &\stackrel{\text{def}}{=} in(x).C'(x) \\ C'(x) &\stackrel{\text{def}}{=} \overline{out}(x).C \end{aligned}$$

For each value v, $in(x)$ becomes in_v and $\overline{out}(x)$ becomes $\overline{out}_v$.

Thus,

$$\begin{aligned} C &\stackrel{\text{def}}{=} \Sigma_{v\in V}\, in_v.C'_v \\ C'_v &\stackrel{\text{def}}{=} \overline{out}_v.C(v \in V) \end{aligned}$$

EX

$$F \equiv in(x).(\text{if } x < 5 \text{ then } \overline{out}(x).\mathbf{0})$$

is translated to

$$\hat{F} \equiv \sum_{v\in V} E_v$$

where

$$E_v \equiv \begin{cases} in_v.\overline{out}_v.\mathbf{0} & \text{if } v < 5 \\ \mathbf{0} & \text{otherwise} \end{cases}$$

Recursive Expressions

$A \stackrel{\text{def}}{=} a.A$ is not allowed in constant expression.

$\mathbf{fix}(X = a.X)$: the agent X such that $X \stackrel{\text{def}}{=} a.X$

To define the recursive expression A, $A \stackrel{\text{def}}{=} \mathbf{fix}(X = a.X)$

(6) $\mathbf{fix}_j(\{X_i = E_i : i \in I\})$, a Recursion ($I$ an indexing set and $j \in I$)

Briefly,

$$\mathbf{fix}_j(\tilde{X} = \tilde{E})$$

For a *single* equation, we write $\mathbf{fix}(X = E)$.

A new inference rule for the simple case of a single equation,

$$\textbf{Rec}\ \frac{E\{\mathbf{fix}(X = E)/X\} \xrightarrow{\alpha} E'}{\mathbf{fix}(X = E) \xrightarrow{\alpha} E'}$$

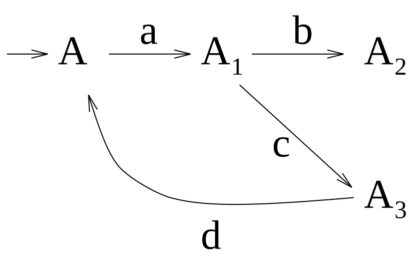
A transition rule for the full form $\mathbf{fix}_j(\tilde{X} = \tilde{E})$,

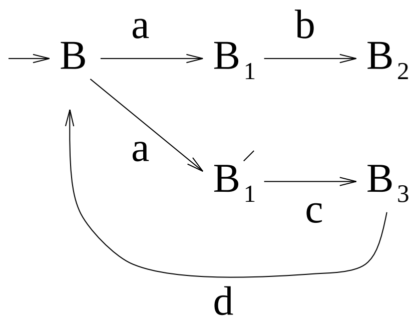
$$\textbf{Rec}_j\ \frac{E_j\{\widetilde{\mathbf{fix}}(\tilde{X} = \tilde{E})/\tilde{X}\} \xrightarrow{\alpha} E'}{\mathbf{fix}_j(\tilde{X} = \tilde{E}) \xrightarrow{\alpha} E'}$$

2 Strong Bisimulation and Strong Equivalence

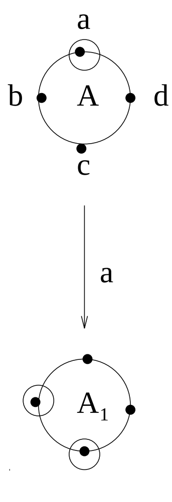
EX

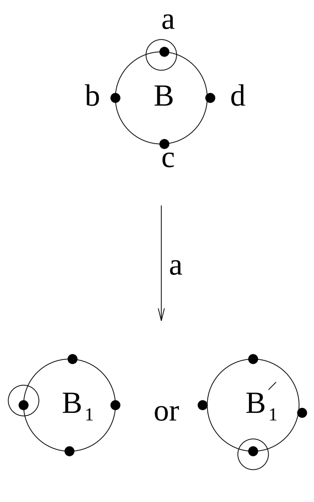
$$\begin{array}{ll}
A \stackrel{\text{def}}{=} a.A_1 & B \stackrel{\text{def}}{=} a.B_1 + a.B_1' \\
A_1 \stackrel{\text{def}}{=} b.A_2 + c.A_3 & B_1 \stackrel{\text{def}}{=} b.B_2 \qquad B_1' \stackrel{\text{def}}{=} c.B_3 \\
A_2 \stackrel{\text{def}}{=} \mathbf{0} & B_2 \stackrel{\text{def}}{=} \mathbf{0} \\
A_3 \stackrel{\text{def}}{=} d.A & B_3 \stackrel{\text{def}}{=} d.B
\end{array}$$



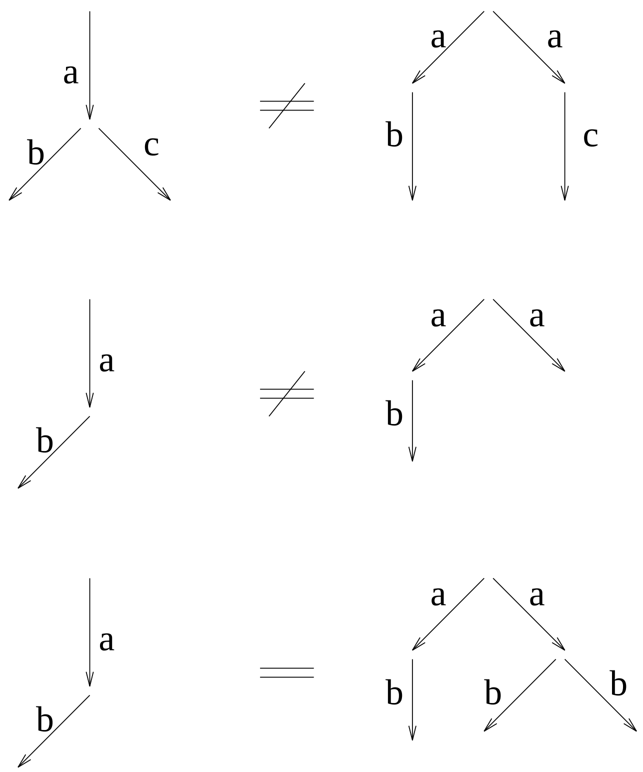


$(a.c.d)^*.a.b.\mathbf{0}$ $\qquad$ $(a.c.d)^*.a.b.\mathbf{0}$





$$a.(P+Q) \neq a.P + a.Q$$



A notion of equivalence between agents.

Treat τ exactly like any other action.

P and Q are equivalent iff, for every action α, every α-derivative of P is equivalent to some α-derivative of Q, and conversely.

Formally, ($\sim$ for equivalence relation)

$(*)$ $P \sim Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \xrightarrow{\alpha} Q'$ and $P' \sim Q'$

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

for some P', $P \xrightarrow{\alpha} P'$ and $P' \sim Q'$

Note

(1) It is not a definition since there are many relations which satisfy it (*e.g.*, identity relation).

(2) $P|Q \sim Q|P$, $P + \mathbf{0} \sim P$, *etc.*

(3) We want the largest (or weakest) relation $\sim$.

Def A binary relation $\mathcal{S} \subseteq P \times P$ over agents is a *strong bisimulation* if $(P, Q) \in \mathcal{S}$ implies, for all $\alpha \in Act$,

(i) Whenever $P \stackrel{\alpha}{\to} P'$ then,

for some Q', $Q \stackrel{\alpha}{\to} Q'$ and $(P', Q') \in \mathcal{S}$

(ii) Whenever $Q \stackrel{\alpha}{\to} Q'$ then,

for some P', $P \stackrel{\alpha}{\to} P'$ and $(P', Q') \in \mathcal{S}$

Note :

Any relation $\sim$ which satisfies $(*)$ is a strong bisimulation.

There are many strong bisimulation relations including $\mathcal{S} = \emptyset$.

EX $(A|B)\backslash c$ and C_1

$$\begin{array}{rcl} \mathcal{S} & = & \{ \ ((A|B)\backslash c, \ C_1), \\ & & \quad ((A'|B)\backslash c, \ C_3), \\ & & \quad ((A|B')\backslash c, \ C_0), \\ & & \quad ((A'|B')\backslash c, \ C_2) \ \} \end{array}$$

Proposition 1 Assume each $\mathcal{S}_i (i = 1, 2, \ldots)$ is a strong bisimulation. Then, the following relations are all strong bisimulation.

$$\begin{array}{ll} (1) Id_{\mathcal{P}} & \qquad (3) \mathcal{S}_1 \mathcal{S}_2 \\ (2) \mathcal{S}_i^{-1} & \qquad (4) \cup_{i \in I} \mathcal{S}_i \end{array}$$

Def P and Q are *strongly equivalent* or *strongly bisimilar*, written $P \sim Q$, if $(P, Q) \in \mathcal{S}$ for some strong bisimulation $\mathcal{S}$. That is,

$$\sim = \cup \{\mathcal{S} : \mathcal{S} \text{ is a strong bisimulation}\}$$

Proposition 2

(1) $\sim$ is the largest strong bisimulation.

(2) $\sim$ is an equivalence relation.

Def $P \sim' Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \stackrel{\alpha}{\Rightarrow} P'$ then,

for some Q', $Q \stackrel{\alpha}{\Rightarrow} Q'$ and $P' \sim Q'$

(ii) Whenever $Q \stackrel{\alpha}{\Rightarrow} Q'$ then,

for some P', $P \stackrel{\alpha}{\Rightarrow} P'$ and $P' \sim Q'$

Proposition 4 $P \sim Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \stackrel{\alpha}{\Rightarrow} P'$ then,

for some Q', $Q \stackrel{\alpha}{\Rightarrow} Q'$ and $P' \sim Q'$

(ii) Whenever $Q \stackrel{\alpha}{\Rightarrow} Q'$ then,

for some P', $P \stackrel{\alpha}{\Rightarrow} P'$ and $P' \sim Q'$

$\sim\mathcal{S}\sim$: a composition of binary relations, so that $P \sim\mathcal{S}\sim Q$ means that for some P' and Q' we have $P \sim P'$, $P'\mathcal{S}Q'$ and $Q' \sim Q$.

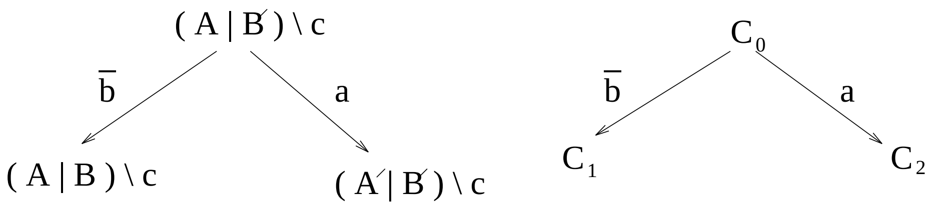
Def $\mathcal{S}$ is a *strong bisimulation up to* $\sim$, if $P\mathcal{S}Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \xrightarrow{\alpha} Q'$ and $P' \sim\mathcal{S}\sim Q'$

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

for some P', $P \xrightarrow{\alpha} P'$ and $P' \sim\mathcal{S}\sim Q'$



Lemma 5 If $\mathcal{S}$ is a strong bisimulation up to $\sim$, then $\sim\mathcal{S}\sim$ is a strong bisimulation.

Proposition 6 If $\mathcal{S}$ is a strong bisimulation up to $\sim$ then $\mathcal{S} \subseteq \sim$.

Hence, to prove $P \sim Q$, we only have to find a strong bisimulation up to $\sim$ which contains (P, Q).

Some Properties of Strong Equivalence

Proposition 7

(1) $P + Q \sim Q + P$

(2) $P + (Q + R) \sim (P + Q) + R$

(3) $P + P \sim P$

(4) $P + \mathbf{0} \sim P$

Proposition 9 : The Expansion Law

Let $P \equiv (P_1[f_1]| \ldots |P_n[f_n])\backslash L$ with $n \geq 1$. Then

$$P \sim \Sigma\{f_i(\alpha).(P_1[f_1]| \ldots |P_i'[f_i]| \ldots |P_n[f_n])\backslash L \ : \\ P_i \xrightarrow{\alpha} P_i', f_i(\alpha) \notin L \cup \bar{L}\} \\ + \Sigma\{\tau.(P_1[f_1]| \ldots |P_i'[f_i]| \ldots |P_j'[f_j]| \ldots |P_n[f_n])\backslash L \ : \\ P_i \xrightarrow{l_1} P_i', P_j \xrightarrow{l_2} P_j', f_i(l_1) = \overline{f_j(l_2)}, i < j\}$$

In the special case (no Relabelling or Reduction),

If $P \equiv P_1| \ldots |P_n$ with $n \geq 1$, then

$$P \sim \Sigma\{\alpha.(P_1| \ldots |P_i'| \ldots |P_n) \ : 1 \leq i \leq n, P_i \xrightarrow{\alpha} P_i'\} \\ + \Sigma\{\tau.(P_1| \ldots |P_i'| \ldots |P_j'| \ldots |P_n) \ : \\ 1 \leq i < j \leq n, P_i \xrightarrow{l_1} P_i', P_j \xrightarrow{l_2} P_j'\}$$

EX

$$A \stackrel{\text{def}}{=} a.A' \qquad B \stackrel{\text{def}}{=} c.B'$$
$$A' \stackrel{\text{def}}{=} \bar{c}.A \qquad B' \stackrel{\text{def}}{=} \bar{b}.B$$

$$\begin{aligned} A|B &= a.A'|c.B' \\ &= a.(\bar{c}.A|c.B') + c.(a.A'|\bar{b}.B) \end{aligned}$$

$$\begin{aligned} \bar{c}.A|c.B' &= \bar{c}.(A|c.B') + c.(\bar{c}.A|B') + \tau.(A|B') \\ & \quad \ldots \end{aligned}$$

Strong Congruence

Proposition 10 Let $P_1 \sim P_2$. Then

(1) $\alpha.P_1 \sim \alpha.P_2$

(2) $P_1 + Q \sim P_2 + Q$

(3) $P_1|Q \sim P_2|Q$

(4) $P_1\backslash L \sim P_2\backslash L$

(5) $P_1[f] \sim P_2[f]$

Def Let E and F contain variables $\tilde{X}$ at most. Then $E \sim F$ if, for all indexed sets $\tilde{P}$ of agents, $E\{\tilde{P}/\tilde{X}\} \sim F\{\tilde{P}/\tilde{X}\}$

"Definition by recursion preserves strong equivalence."

If $A \stackrel{\text{def}}{=} E\{A/X\}, B \stackrel{\text{def}}{=} F\{B/X\}$ and $E \sim F$, then $A \sim B$.

EX

$$\text{Given} \quad \begin{aligned} A &\stackrel{\text{def}}{=} b.\mathbf{0} + a.(A|c.\mathbf{0}) \\ B &\stackrel{\text{def}}{=} a.(c.\mathbf{0}|B) + b.\mathbf{0}, \end{aligned}$$

$$\text{let} \quad \begin{aligned} A &\stackrel{\text{def}}{=} E\{A/X\} \text{ where } E \equiv b.\mathbf{0} + a.(X|c.\mathbf{0}) \\ B &\stackrel{\text{def}}{=} F\{B/X\} \text{ where } F \equiv a.(c.\mathbf{0}|X) + b.\mathbf{0}. \end{aligned}$$

Then $E \sim F$. Therefore, $A \sim B$.

Proposition 11 If $\tilde{X} \stackrel{\text{def}}{=} \tilde{P}$, then $\tilde{X} \sim \tilde{P}$.

Proposition 12 ($\sim$ is preserved by recursive definition)
Let $\tilde{E}$ and $\tilde{F}$ contain variables $\tilde{X}$ at most. Let
$\tilde{A} \stackrel{\text{def}}{=} \tilde{E}\{\tilde{A}/\tilde{X}\}, \tilde{B} \stackrel{\text{def}}{=} \tilde{F}\{\tilde{B}/\tilde{X}\}$ and $\tilde{E} \sim \tilde{F}$, then $\tilde{A} \sim \tilde{B}$.

Unique solution of equations

"There is a unique P (up to $\sim$) such that $P \sim E\{P/X\}$."

This cannot be true for all E. ($e.g.$, $E \equiv X$)

Def X is *weakly guarded* in E if each occurrence of X is within some subexpression $\alpha.F$ of E.

EX X is weakly guarded, but not guarded in $\tau.X$.

Lemma 13 If the variables $\tilde{X}$ are weakly guarded in E, and $E\{\tilde{P}/\tilde{X}\} \xrightarrow{\alpha} P'$, then P' takes the form $E'\{\tilde{P}/\tilde{X}\}$ (for some expression E'), and moreover, for any $\tilde{Q}$,
$E\{\tilde{Q}/\tilde{X}\} \xrightarrow{\alpha} E'\{\tilde{Q}/\tilde{X}\}$.

Proposition 14

(1) If $\tilde{A} \stackrel{\text{def}}{=} \tilde{P}$, then $\tilde{A} \sim \tilde{P}$.

(2) Let the expressions $E_i (i \in I)$ contain at most the variables $X_i (i \in I)$, and let each $X_j (j \in I)$ be weakly guarded in each E_i. Then

$$\text{If } \tilde{P} \sim \tilde{E}\{\tilde{P}/\tilde{X}\} \text{ and } \tilde{Q} \sim \tilde{E}\{\tilde{Q}/\tilde{X}\} \text{ then } \tilde{P} \sim \tilde{Q}.$$

3 Bisimulation and Observation Equivalence

Def If $t \in Act*$, then $\hat{t} \in \mathcal{L}$ is the sequence resulted from deleting all occurrences of τ from t.

Note : $\hat{\tau^n} = \epsilon$.

Def If $t = \alpha_1 \cdots \alpha_n \in Act^*$, then we write $E \xrightarrow{t} E'$ if $E \xrightarrow{\alpha_1} \cdots \xrightarrow{\alpha_n} E'$. We write $E \xrightarrow{t}$ to mean that $E \xrightarrow{t} E'$ for some E'.

A (new) labelled transition system

$$(\mathcal{E}, \mathcal{L}^*, \{\stackrel{s}{\Rightarrow} : s \in \mathcal{L}^*\})$$

Def If $t = \alpha_1 \cdots \alpha_n \in Act^*$, then $E \stackrel{t}{\Rightarrow} E'$ if

$$E(\xrightarrow{\tau})^* \xrightarrow{\alpha_1} (\xrightarrow{\tau})^* (\xrightarrow{\tau})^* \xrightarrow{\alpha_n} (\xrightarrow{\tau})^* E'$$

$E \stackrel{t}{\Rightarrow}$ means $E \stackrel{t}{\Rightarrow} E'$ for some E'.

Note : $(\xrightarrow{\tau})^*$ means "zero or more τ actions".

Def If $t \in Act^*$, then E' is a t-*descendent* of E iff $E \stackrel{\hat{t}}{\Rightarrow} E'$.

Note :

$E \stackrel{ab}{\Rightarrow} E'$ means that $E \stackrel{\tau^p}{\rightarrow}\stackrel{a}{\rightarrow}\stackrel{\tau^q}{\rightarrow}\stackrel{b}{\rightarrow}\stackrel{\tau^r}{\rightarrow} E'$ for some $p, q, r \geq 0$.

$E \stackrel{\epsilon}{\Rightarrow} E'$ iff $E \stackrel{\tau^n}{\rightarrow} E'$ for some $n \geq 0$.

E' is a τ-descendent of E iff $E \stackrel{\tau^n}{\rightarrow} E'$ for some $n \geq 0$.

Difference between three relations $\stackrel{t}{\rightarrow}$, $\stackrel{t}{\Rightarrow}$ and $\stackrel{\hat{t}}{\Rightarrow}$, for $t \in Act^*$:

$\stackrel{t}{\rightarrow}$ specifies *exactly* the τ actions occurring in t.

$\stackrel{t}{\Rightarrow}$ specifies *at least* the τ actions occurring in t.

$\stackrel{\hat{t}}{\Rightarrow}$ specifies nothing about τ actions.

$E \stackrel{\tau}{\Rightarrow} E$, $\qquad\qquad$ $E \stackrel{\tau}{\not\Rightarrow} E$,

$E \stackrel{\hat{\tau}}{\Rightarrow} E$ $\qquad\qquad$ $E \stackrel{\hat{\tau}}{\Rightarrow} E$

Thus $P \stackrel{t}{\rightarrow} P'$ implies $P \stackrel{t}{\Rightarrow} P'$, and $P \stackrel{t}{\Rightarrow} P'$ implies $P \stackrel{\hat{t}}{\Rightarrow} P'$.

P and Q are observation-equivalent iff, for every action α, every α-derivative of$P$ is observation-equivalent to some α-descendant of Q, and similary with P and Q interchanged.

Formally, ($\approx$ for observation equivalence)

$(*)$ $P \approx Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \stackrel{\hat{\alpha}}{\Rightarrow} Q'$ and $P' \approx Q'$

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

for some P', $P \stackrel{\hat{\alpha}}{\Rightarrow} P'$ and $P' \approx Q'$

Def A binary relation $\mathcal{S} \subseteq P \times P$ over agents is a *(weak) bisimulation* if $(P, Q) \in \mathcal{S}$ implies, for all $\alpha \in Act$,

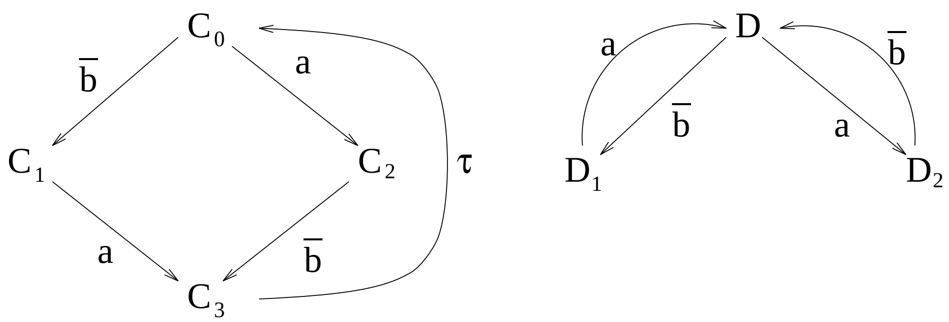
(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \stackrel{\hat{\alpha}}{\Rightarrow} Q'$ and $(P', Q') \in \mathcal{S}$

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

for some P', $P \stackrel{\hat{\alpha}}{\Rightarrow} P'$ and $(P', Q') \in \mathcal{S}$

EX



$$\mathcal{S} = \{(C_0, D), (C_1, D_1), (C_2, D_2), (C_3, D)\}$$

is a bisimulation, though there can be no strong bisimulation containing the pair (C_3, D).

Note : If we had used $\stackrel{\alpha}{\Rightarrow}$ instead of $\stackrel{\hat{\alpha}}{\Rightarrow}$ then there would be no bisimulation containing (C_3, D).

$$(C_3, D), C_3 \xrightarrow{\tau} C_0, D \stackrel{\tau}{\Rightarrow} D' \text{: impossible}$$

EX $P \equiv a.\mathbf{0} + b.\mathbf{0} \qquad Q \equiv a.\mathbf{0} + \tau.b.\mathbf{0}$

(P, Q) cannot be in any bisimulation, assuming $a \neq b$.

$Q \xrightarrow{\tau} b.\mathbf{0}$; $b.\mathbf{0} \overset{a}{\not\rightarrow}$, whereas, $P \overset{\epsilon}{\Rightarrow} P$; $P \xrightarrow{a} \mathbf{0}$.

EX

$$
\begin{array}{ll}
A_0 \stackrel{\text{def}}{=} a.A_0 + b.A_1 + \tau.A_1 & \\
A_1 \stackrel{\text{def}}{=} a.A_1 + \tau.A_2 & B_1 \stackrel{\text{def}}{=} a.B_1 + \tau.B_2 \\
A_2 \stackrel{\text{def}}{=} b.A_0 & B_2 \stackrel{\text{def}}{=} b.B_1
\end{array}
$$

$\mathcal{S} = \{(A_0, B_1), (A_1, B_1), (A_2, B_2)\}$ is a bisimulation.

Def P and Q are *observation-equivalent* or *(weakly) bisimilar*, written $P \approx Q$, if $(P, Q) \in \mathcal{S}$ for some (weak) bisimulation $\mathcal{S}$. That is,

$$\approx = \cup\{\mathcal{S} : \mathcal{S} \text{ is a bisimulation}\}$$

Proposition 2

(1) $\approx$ is the largest strong bisimulation.

(2) $\approx$ is an equivalence relation.

Proposition 4 $P \approx Q$ iff, for all $\alpha \in Act$,

(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \stackrel{\hat{\alpha}}{\Rightarrow} Q'$ and $P' \approx Q'$

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

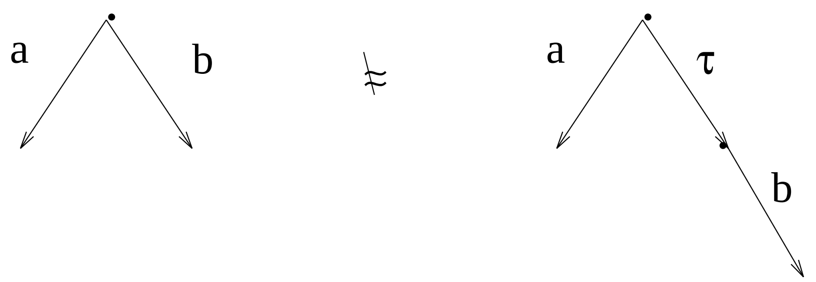
for some P', $P \stackrel{\hat{\alpha}}{\Rightarrow} P'$ and $P' \approx Q'$

Some properties of bisimilarity

Proposition 7 $P \approx \tau.P$

EX $b.\mathbf{0} \approx \tau.b.\mathbf{0}$

However, $a.\mathbf{0} + b.\mathbf{0} \not\approx a.\mathbf{0} + \tau.b.\mathbf{0}$



Note : $\approx$ is not a congruence.

Proposition 10 If $P \approx Q$, then $\alpha.P = \alpha.Q$.

Proposition 11 If $P \approx Q$, then

$$P|R \approx Q|R, P\backslash L \approx Q\backslash L, P[f] \approx Q[f]$$

.

4 Observation Congruence

A notion of *experiment* :

$P \stackrel{s}{\Rightarrow} P'$, where $s \in \mathcal{L}^*$, is an experiment upon P.

If $s = \epsilon$, we write $P \Rightarrow P'$.

EX $P \equiv a.\mathbf{0} + \tau.\mathbf{0}$

P admits a-experiment.

But $P \Rightarrow \mathbf{0}$ does not admit an a-experiment.

Proposition 1 $\mathcal{S}$ is a bisimulation iff, for all $(P,\ Q) \in \mathcal{S}$ and $s \in \mathcal{L}^*$,

(i) Whenever $P \stackrel{s}{\Rightarrow} P'$ then,

for some $Q',\ Q \stackrel{s}{\Rightarrow} Q'$ and $P'\mathcal{S}Q'$.

(ii) Whenever $Q \stackrel{s}{\Rightarrow} Q'$ then,

for some $P',\ P \stackrel{s}{\Rightarrow} P'$ and $Q'\mathcal{S}P'$.

Why the empty expression $P \Rightarrow P'$, with $s = \epsilon$, must be allowed:

Without it, $a.\mathbf{0} + \tau.\mathbf{0}$ would be bisimilar to $a.\mathbf{0}$

In general, $\approx$ not preserved by Summation.

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

$$a.\mathbf{0} + b.\mathbf{0} \not\approx a.\mathbf{0} + \tau.b.\mathbf{0}$$

Proposition 2 Bisimilarity is preserved by Composition, Restriction, and Relabelling; that is, if $P \approx Q$ then $P|R \approx Q|R$, $P\backslash L \approx Q\backslash L$ and $P[f] \approx Q[f]$.

Note: $\approx$ not fully substitutive;

$$P \approx Q \text{ does not imply } P + R \approx Q + R.$$

We want $P = Q$, which implies $P \approx Q$ and is fullly substitutive.

It should be the largest congruence relation included in $\approx$.

Def P and Q are *equal* or *(observation-)congruence*, $P = Q$, if for all α

(i) Whenever $P \xrightarrow{\alpha} P'$ then,

for some Q', $Q \overset{\alpha}{\Rightarrow} Q'$ and $P' \approx Q'$.

(ii) Whenever $Q \xrightarrow{\alpha} Q'$ then,

for some P', $P \overset{\alpha}{\Rightarrow} P'$ and $Q' \approx P'$.

Note : The two clauses differ from $\approx$: $\overset{\alpha}{\Rightarrow}$ in place of $\overset{\hat{\alpha}}{\Rightarrow}$.

Thus, $P + Q$ must be matched by *at least* one action of the other. This applies only to the first actions of P and Q, since $P' \approx Q'$, not $P' = Q'$, for their corresponding derivatives.

Proposition 4 $P \sim Q$ implies $P = Q$, and $P = Q$ implies $P \approx Q$.

EX Give simple examples of P and Q for which

(1) $P = Q$ but $P \not\sim Q$

(2) $P \approx Q$ but $P \neq Q$.

Proposition 5 Equality is an equivalence relation.

Proposition 7 If $P = Q$, then $\alpha.P = \alpha.Q$, $P + R = Q + R$, $P|R = Q|R$, $P\backslash L = Q\backslash L$ and $P[f] = Q[f]$.

Axioms for finite agents

Def An agent expression is *finite* if it contains only finite Summations, and no Constants (or Recursions).

Note: By use expansion law, any finite agent can be equated to one which contains no Composition, Restriction or Relabelling.

The Axiom System $\mathcal{A}^f$ and $\mathcal{A}$

Axioms $\mathcal{A}_1$

S1 $P + Q = Q + P$

S2 $P + (Q + R) = (P + Q) + R$

S3 $P + P = P$

S4 $P + \mathbf{0} = P$

Axioms $\mathcal{A}_2$

S1-S4

T1 $\alpha.\tau.P = \alpha.P$

T2 $P + \tau.P = \tau.P$

T3 $\alpha.(P + \tau.Q) + \alpha.Q = \alpha.(P + \tau.Q)$

Def P is *a standard form* if

$$P \equiv \sum_{i=1}^{m} \alpha_i.P_i$$

where each P_i is also in standard form.

Proposition 15 $\mathcal{A}_1$ is sound and complete for $\sim$.

$$P \sim Q \text{ iff } \mathcal{A}_1 \vdash P = Q$$

Def P is *a full standard form* if

(1) $P \equiv \Sigma_{i=1}^{m} \alpha_i.P_i$, where each P_i is in full standard form;

(2) Whenever $P \stackrel{\alpha}{\Rightarrow} P'$ then $P \stackrel{\alpha}{\rightarrow} P'$.

Proposition 18 $\mathcal{A}_2$ is sound and complete for $=$.

$$P = Q \text{ iff } \mathcal{A}_2 \vdash P = Q$$

Axioms for finite-state agents

Def The class of behavior expressions $\mathcal{E}$ is defined by the following syntax: $(X \in Var,\ a \in Act)$

$$E ::= \mathbf{0} \mid X \mid \alpha.E \mid \mathbf{fix}(X = E) \mid E + E$$

The Axiom System $\mathcal{A}_3$

Summation axioms

S1 $E + F = F + E$

S2 $E + (F + G) = (E + F) + G$

S3 $E + E = E$

S4 $E + 0 = E$

τ-Axioms

T1 $\alpha.\tau.E = \alpha.E$

T2 $E + \tau.E = \tau.E$

T3 $\alpha.(E + \tau.F) + \alpha.F = \alpha.(E + \tau.F)$

Recursion Axioms

R1 $\mathbf{fix}(X = E) = E\{\mathbf{fix}(X = E)/X\}$

R2 If $F = E\{F/X\}$ then $F = \mathbf{fix}(X = E)$,

provided X is guarded in E

R3 $\mathbf{fix}(X = X + E) = \mathbf{fix}(X = E)$

R4 $\mathbf{fix}(X = \tau.X + E) = \mathbf{fix}(X = \tau.E)$

R5 $\mathbf{fix}(X = \tau.(X + E) + F) = \mathbf{fix}(X = \tau.X + E + F)$

Proposition (Soundness)

If $\mathcal{A}_3 \vdash E = F$ then $E = F$.

Proposition (Completeness)

If $E = F$ then $\mathcal{A}_3 \vdash E = F$

Bibliography

1. R. Milner, *Communication and Concurrency*, Prentice-Hall, 1989.
2. R. Milner, "A Complete Axiomatization for Observational Congruence of Finite-State Behaviors," *Information and Computation*, 81, pp. 227-247, 1989.