2.1 A clock that may stop after a tick and 'ticks' at least once

$Clock \overset{def}{=} tick.0 + tick.Clock$

$Clock \overset{def}{=} tick.(Clock + 0)$ is wrong. Always ticks.

2.2 A CM that can fail.

$CM \overset{def}{=} coin.CM + coin.\overline{coffee}.CM$

$CM \overset{def}{=} coin.(CM + \overline{coffee}.CM)$ is wrong. Not as wrong as the previous one but leaves the choice to the user.

2.3

$$T_q \overset{def}{=} \sum_{a \in A} a.T_{\delta(a,q)}$$

$$T \overset{def}{=} T_{q_0}$$

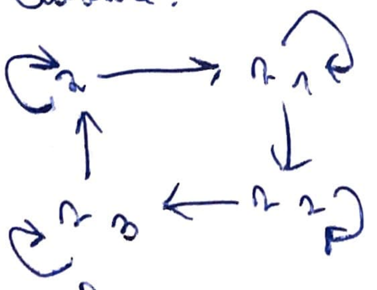
2.4

$Proc = \{r, r_1, r_2, r_3\}$

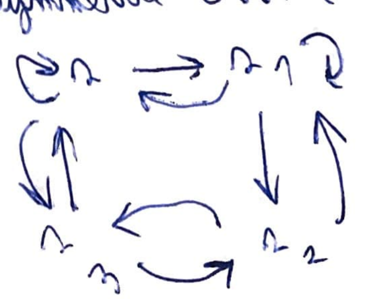
$Act = \{a\}$

$\xrightarrow{a} = \{(r, r_1), (r_1, r_2), (r_2, r_3), (r_3, r)\}$

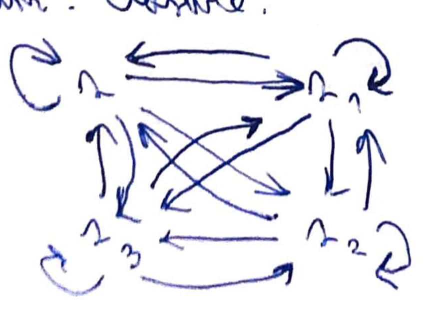
Refl. closure:



Symmetric closure:



Trans. closure:



In this case, the trans. closure is also the reflexive, symmetric and transitive closure.

(2.5)

$p_2 \xrightarrow{0} p_2$ The set of reachable states from p_2 is

$p_2 \xrightarrow{d} p_1$ $\{p, p_1, p_2\}$

$p_2 \xrightarrow{2} p$

(2.6)

$a.b.A + B$ ✓

$(a.0 + \bar{a}.A) \setminus \{a, b\}$ ✓

$(a.0 + \bar{a}.A) \setminus \underbrace{\{a, \tau\}}$ X

$(a.B + [a/b])$ X This must be a labels set. τ is an action.

Bad syntax.

$\tau.\tau.B + 0$ ✓

$(a.B + b.B)[a/b, b/a]$ ✓

$(a.B + \tau.B)[a/\tau, b/a]$ X $f(\tau) = a$ is forbidden. $[\tau/a, b/a]$ is fine, but a/τ is not.

$(a.b.A + \bar{a}.0) \mid B$ ✓

$\underline{(a.b.A + \bar{a}.0)}.B$ X

This is not an action.

$(a.b.A + \bar{a}.0) + B$ ✓

$(0 \mid 0) + 0$ ✓

The counter

$counter_0 \stackrel{def}{=} up.counter_1$

$counter_n \stackrel{def}{=} up.counter_{n+1} + down.counter_{n-1}$

It cannot be feed to a computer. The following can be and has the same "behavior":

$C \stackrel{def}{=} up.(C \mid down.0)$

2.7

$CS \overset{def}{=} \overline{pub}.CS_1$

$CS_1 \overset{def}{=} \overline{coin}.CS_2$

$CS_2 \overset{def}{=} coffee.CS$

$SmUni \overset{def}{=} (CM|CS) \setminus \{coin, coffee\}$

$CM \overset{def}{=} coin.CM_1$

$CM_1 \overset{def}{=} \overline{coffee}.CM$

$\cancel{Clock \overset{def}{=} tick.(Clock+0)}$

$(CS, CS_1) \in \xrightarrow{\overline{pub}}$

$(CS1, CS2) \in \xrightarrow{\overline{coin}}$

$(CS2, CS) \in \xrightarrow{coffee}$

$(CM, CM1) \in \xrightarrow{coin}$

$(CM1, CM) \in \xrightarrow{\overline{coffee}}$

$(CM|CS, CM1|CS) \in \xrightarrow{coin}$

$(CM|CS, CM|CS1) \in \xrightarrow{\overline{pub}}$

$(CM|CS1, CM|CS2) \in \xrightarrow{\overline{coin}}$

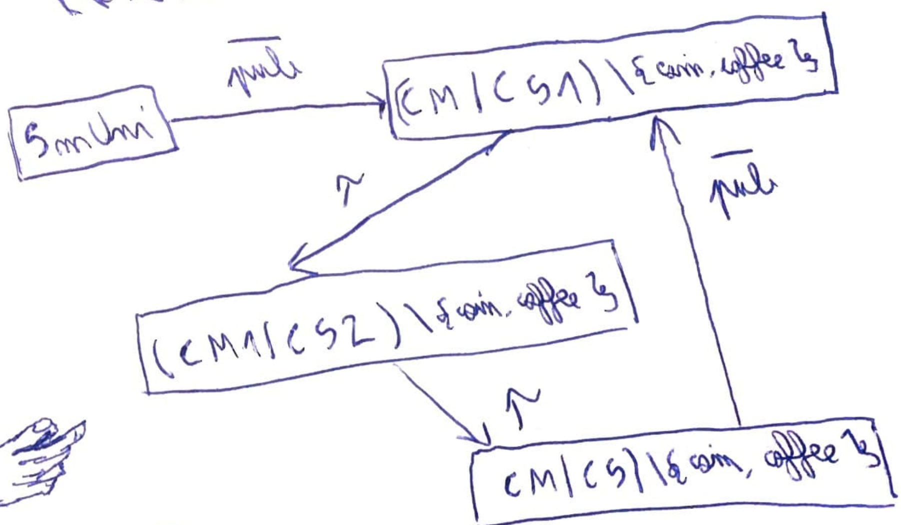
$(CM|CS1, CM1|CS2) \in \xrightarrow{\tau}$

$(CM1|CS, CM1|CS1) \in \xrightarrow{\overline{pub}}$

$(CM1|CS1, CM1|CS2) \in \xrightarrow{\overline{coin}}$

$(CM|CS1, CM1|CS1) \in \xrightarrow{coin}$

$(CM1|CS2, CM|CS) \in \xrightarrow{\tau}$



(2.8) $A \overset{def}{=} b.a.B$

$$\dfrac{\dfrac{\dfrac{}{b.a.B \xrightarrow{b} a.B}\ \text{ACT}}{A \xrightarrow{b} a.B}\ \text{CON} \qquad \dfrac{}{\bar{b}.0 \xrightarrow{\bar{b}} 0}\ \text{ACT}}{\dfrac{A \mid \bar{b}.0 \xrightarrow{\tau} a.B \mid 0}{(A \mid \bar{b}.0)\setminus b \xrightarrow{\tau} (a.B \mid 0)\setminus b}\ \text{RES}}\ \text{COM3}$$

$$\dfrac{\dfrac{\dfrac{}{\bar{b}.a.B \xrightarrow{\bar{b}} a.B}\ \text{ACT}}{A \mid \bar{b}.a.B \xrightarrow{\bar{b}} (A \mid a.B)}\ \text{COM2}}{(A \mid \bar{b}.a.B)_0 + (\bar{b}.A)[a/b]_1 \rightarrow (A \mid a.B)}\ \text{SUM}_0$$

$$\dfrac{\dfrac{\dfrac{}{\bar{b}.A \xrightarrow{\bar{b}} A}\ \text{ACT}}{(\bar{b}.A)[a\setminus b] \xrightarrow{\bar{a}} A[a\setminus b]}\ \text{REL}\quad f(\bar{b})=\bar{a}}{(A \mid \bar{b}.a.B)_0 + (\bar{b}.A)[a\setminus b]_1 \xrightarrow{\bar{a}} A[a\setminus b]}\ \text{SUM}_1$$

(2.9) $A \overset{def}{=} (a_0.A)\setminus b$

$(a_0.A)\setminus b \xrightarrow{a_0} ((a_0.A)\setminus b)\setminus b$

$A \overset{def}{=} (a_0.A\setminus b)$

$a_0.A\setminus b \xrightarrow{a_0} (a_0.A\setminus b)\setminus b$

$A \overset{def}{=} a_0.A$

$(a_0.A) \xrightarrow{a_0} (a_0.A)$ Perform a_0 inf. many times.

(2.10) $A \overset{def}{=} (a_0.A)[f]$ $\qquad f(a_i) = a_{i+1} \quad \forall i \in \mathbb{N}$

$(a_0.A)[f] \xrightarrow{a_1} ((a_0.A)[f])[f] \xrightarrow{a_2} (((a_0.A)[f])[f])[f] \ldots$

In this case the process needs to perform infinitely many different actions.

2.11 MC HOMEW ~~x~~ 2

a) $Mutex_1 \stackrel{def}{=} (User \mid Sem) \setminus p \setminus v$

$User \stackrel{def}{=} \bar{p}.enter.exit.\bar{v}.User$

$Sem \stackrel{def}{=} p.v.Sem$

$Act = \{ \tau, enter, exit, \overline{enter}, \overline{exit} \}$ ~~...~~

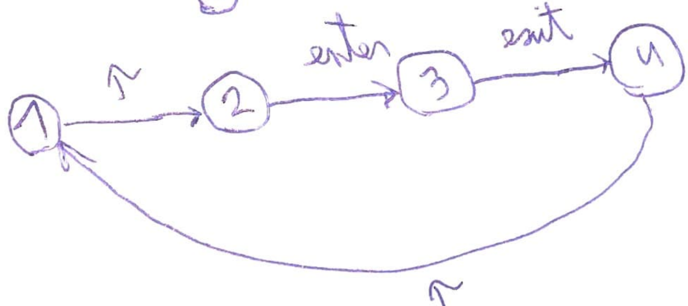
$Proc = \{ (User \mid Sem) \setminus p \setminus v, \ldots ②, ③, ④$

$\xrightarrow{\tau} = \{ ((User \mid Sem) \setminus p \setminus v, (enter.exit.\bar{v}.User \mid v.Sem) \setminus p \setminus v),$ ~~...~~ $\}$

$((\bar{v}.User \mid v.Sem) \setminus p \setminus v,$ ~~...~~ ①

$\xrightarrow{enter} = \{ ((enter.exit.\bar{v}.User \mid v.Sem), (exit.\bar{v}.User \mid v.Sem) \setminus p \setminus v)\}$

$\xrightarrow{exit} = \{ ((exit.\bar{v}.User \mid v.Sem) \setminus p \setminus v, (\bar{v}.User \mid v.Sem) \setminus p \setminus v)\}$



b) $Mutex\,2 \stackrel{def}{=} ((User \mid Sem) \mid User) \setminus p \setminus v$

It would be different because both users could be in state

exit . User

$((exit.User \mid p.v.Sem) \mid exit.User) \setminus p \setminus v$

c) $FUser \stackrel{def}{=} \bar{p}.enter.(exit.\bar{v}.FUser + exit.\bar{v}.0)$

$FMutex \stackrel{def}{=}$ ~~enter exit~~ $((User \mid Sem) \mid FUser) \setminus p \setminus v$

The sequence of performed actions will be the same.

2.12 MC. Hamew 4

FIFO

$FIFO \overset{def}{=} in(x).(\overline{out}(x).FIFO + in(y).(\overline{out}(y) \ldots$ X $FIFO \overset{def}{=} in(x).FIFO_1(x)$

$FIFO_1(x) \overset{def}{=} \overline{out}(x).FIFO + in(y).FIFO_2(x,y)$

$FIFO_2(x,y) = \overline{out}(x).FIFO_1(y) + \overline{out}(y).FIFO_1(x)$

¿How to model it by means of Cell?

$BAG \overset{def}{=} in(x).BAG(x)$

$BAG(x) \overset{def}{=} \overline{out}(x).0 \mid in(x).BAG(x)$

Is this legal (fix the x).

$BAG \mid BAG$

¿How to model it by means of Cell?

2.13

$\cancel{B \xrightarrow{empty?} B}$ ¿¿??

$\cancel{B \xrightarrow{push(1)} (push(y))}$

3.1

I is reflexive but nothing more.

U is an equiv. relation. Weak because it does not tell appart anything.

$\leq$ is a preorder only.

M_2 is an equivalence relation.

(3.2)

1. $CTS((CA|CTM) \setminus \{coin, coffee, tea\}) = \{\}$

$CTS((CA|CTM') \setminus \{coin, coffee, tea\}) = \{(\tau), (\tau\tau), (\tau\tau\tau) \dots\}$

2. ~~Let $\alpha_1 \dots \alpha_k \in CTS(P) = CTS(Q)$~~

~~Let $L \subseteq Act$~~

~~If $\alpha_i \in L \wedge \alpha_j \notin L \ \forall j < i$:~~

~~$\alpha_1 \dots \alpha_i \in CTS(P \setminus L)$~~

~~$\in CTS(Q \setminus L)$~~

~~else if $\alpha_i \notin L \ \forall i$~~

~~$\alpha_1 \dots \alpha_k \in CTS(P \setminus L)$~~

~~$\in CTS(Q \setminus L)$~~

No, counterexample:

~~$CTS((CA|CTM) \setminus \{tea\}) = CTS((CA|CTM') \setminus \{tea\}) = \emptyset$~~

$A = a.0$

$A' = a.0 + b.c.A''$

$A'' = a.A''$

$CTS(A) = \{a\}$

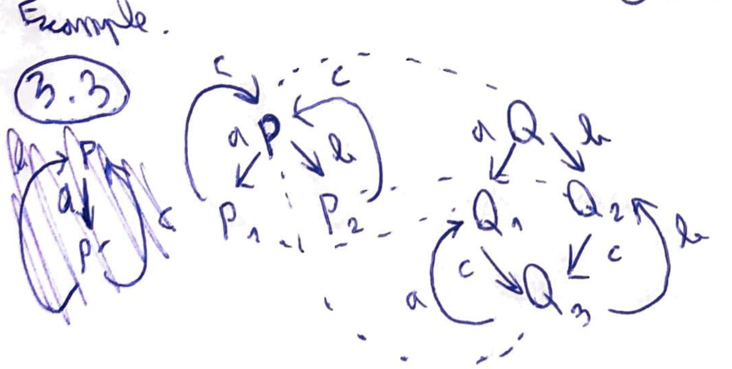
$CTS(A') = \{a\}$

$CTS(A \setminus c) = \{a\}$

$CTS(A' \setminus c) = \{a, b\}$

Example.

(3.3)



(P, Q)

P can perf. a (P1, Q1)

P can perf. b (P2, Q2)

Same from Q.

(P1, Q1)

Both can perf. c : (P, Q3)

(P2, Q2)

" : (P, Q3)

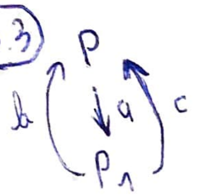
(P, Q3)

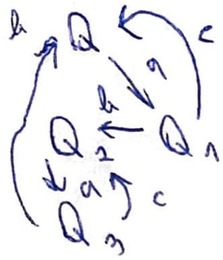
Both can perf a : (P1, Q1)

" b : (P2, Q2)

ML. HOMEW 3

(3.3)





(P, Q):

$a: (P_1, Q_1)$

(P_1, Q_1):

$b: (P, Q_2)$

$c: (P, Q)$

(P, Q_2):

$a: (P_1, Q_3)$

(P_1, Q_3):

$b: (P, Q)$

$c: (P, Q_2)$

$R = \{(P,Q), (P_1,Q_1), (P,Q_2), (P_1,Q_3)\}$

(3.4)

$P = a.P_1$

$P_1 = b.0 + c.0$

$Q = a.b.0 + a.c.0$

$Q_1 = b.0$

$Q_2 = c.0$

Reasoning in by contr.

Suppose $\exists R$: $(P,Q) \in R$

$\Rightarrow (b.0 + c.0, \underbrace{b.0}_{\text{cannot perf. } c}) \in R$

contradiction.

(P, Q):

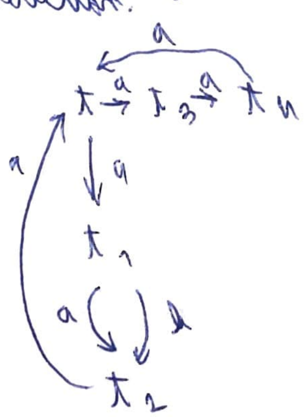
$a: (P_1, Q_1)$

$b: (P_1, Q_2)$

(P_1, Q_1):

P_1 can perf. c but Q_1 cannot.

(3.5)



(s, t):

$\cancel{a: (s_1, t_2)}$

$a: (s_2, t_3)$ ✓

$a: (s_1, t_1)$ ✓

$\cancel{a: (s_2, t_1)}$

$\cancel{(s_1, t_3)}$:

(s_2, t_3):

$a: (s_4, t_4)$

(s_4, t_4):

$a: (s, t)$

(s_1, t_1):

$a: (s_3, t_2)$

$b: (s_4, t_2)$

(s_3, t_2)

$a: (s, t)$

(s_4, t_2)

$a: (s, t)$

$R = \{(s,t), (s_2,t_3), (s_1,t_1), (s_4,t_4), (s_1,t_1), (s_3,t_2), (s_4,t_2)\}$

3.6 Given an LTS

$s \sim s$) $I = \{(s,s) \mid s \in Proc\}$ is a bisimulation: (strong)

Let $(s,s) \in I$

If $s \xrightarrow{\alpha} s'$, then $s \xrightarrow{\alpha} s'$ and $(s',s') \in I$

$s_1 \sim s_2 \Rightarrow s_2 \sim s_1$

If $s_1 \sim s_2$, $\exists R$ s.t. $(s_1,s_2) \in R$

Consider $R^{-1} = \{(s',s) \mid (s,s') \in R\}$

As $(s_1,s_2) \in R$, then $(s_2,s_1) \in R$

And R is a bisimulation:

If $(s_1', s_2') \in R^{-1}$, as $(s_2', s_1') \in R$

If $s_1' \xrightarrow{\alpha_1} s_1''$, then $\exists\, s_2' \xrightarrow{\alpha_1} s_2''$ and $(s_2'', s_1'') \in R \Rightarrow (s_1'', s_2'') \in R^{-1}$

If $s_2' \xrightarrow{\alpha_2} s_2'''$, then $\exists\, s_1' \xrightarrow{\alpha_2} s_1'''$ and $(s_2''', s_1''') \in R \Rightarrow (s_1''', s_2''') \in R$

$s_1 \sim s_2 \wedge s_2 \sim s_3 \Rightarrow s_1 \sim s_3$

$\exists R_1$ s.t. $(s_1,s_2) \in R_1$

$\exists R_2$ s.t. $(s_2,s_3) \in R_2$

Consider $R = \{(s_1', s_3') \mid (s_1', s_2') \in R_1 \wedge (s_2', s_3') \in R_2\}$

As $(s_1,s_2) \in R_1 \wedge (s_2,s_3) \in R_2$, $(s_1,s_3) \in R$.

And R is strong bisim.:

If $(s_1', s_3') \in R$, then $\exists\, s_2'$: $(s_1', s_2') \in R_1$, $(s_2', s_3') \in R_3$

If $s_1' \xrightarrow{\alpha} s_1''$:

$s_2' \xrightarrow{\alpha} s_2''$ and $(s_1'', s_2'') \in R_1$

$s_3' \xrightarrow{\alpha} s_3''$ and $(s_2'', s_3'') \in R_2$ $\Rightarrow (s_1'', s_3'') \in R$

If $s_1' \xrightarrow{\beta} s_1'''$:

$s_2' \xrightarrow{\beta} s_2'''$ and $(s_1''', s_2''') \in R_1$

$s_3' \xrightarrow{\beta} s_3'''$ and $(s_2''', s_3''') \in R_2 \Rightarrow (s_1''', s_3''') \in R$

... HOMEW 4

(3.9)

(s,t) are strong bisimilar $\Leftrightarrow$ (s,t) are string bisimilar

($\Rightarrow$) If (s,t) are strong bis. $\exists R$ bisimulation s.t. $(s,t) \in R$
~~Consider~~ (*)
Induction over σ.

* Un poco mal, mejor considerar la clausura reflexiva y transitiva de R.

If $s \xrightarrow{\alpha_1} s'$
$\exists\, t \xrightarrow{\alpha_1} t' : (s',t') \in R \Rightarrow (\alpha_1) \in R'$

If $t \xrightarrow{\alpha_2} t''$
$\exists\, s \xrightarrow{\alpha_2} s'' : (s'',t'') \in R \Rightarrow (\alpha_2) \in R'$

(*) Consider $R' = \{(\alpha_1, \alpha_2, \dots, \alpha_m) \mid (\alpha_i, \alpha_{i+1}) \in R \ \forall i \in \{1,2,\dots,m-1\}\}$

Let $\sigma = (\alpha_1, \alpha_2, \dots, \alpha_m) \in R'$, $\sigma' = (\alpha_2, \alpha_3, \dots, \alpha_m)$

~~If $s \xrightarrow{\sigma} s_1'$ then $\exists s_1''$ s.t. $s \xrightarrow{\alpha_1} s_1'' \xrightarrow{\sigma'} s_1'$ and by induction hypothesis: $\exists t_1''$ s.t. $t \xrightarrow{\sigma'} t_1''$ with $(s_1'', t_1'') \in R'$ and $(s_1'', t_1'') \in R$~~

If $s \xrightarrow{\sigma} s_1'$ then $\exists s_1''$ s.t.
$s \xrightarrow{\alpha_1} s_1'' \xrightarrow{\sigma'} s_1'$

As $(s,t) \in R$, $\exists t_1''$ s.t. $t \xrightarrow{\alpha_1} t_1''$
with $(s_1'', t_1'') \in R$ and, by induct:
$\exists t_1'$ s.t. $t_1'' \xrightarrow{\sigma'} t_1'$, with $(s_1', t_1') \in R'$,
so, $(s,t) \in R'$.

Same for $t \xrightarrow{\sigma} t_1'$.

($\Leftarrow$) Is easy considering sequences of length 1.
The same strong bisimulation serves also as a strong bisimulation.

MC. Hamew 5

3.11

If $P \sim Q$, then P and Q are also strong bisimilar.

Let $\sigma_1 = \alpha_1 \dots \alpha_{k_1} \in$ Traces P such that $P \xrightarrow{\sigma_1} P'$
As they are strong bis. $\exists Q'$ s.t $Q \xrightarrow{\sigma_1} Q'$,
so $\sigma_1 \in$ Traces (Q)

Let $\sigma_2 = \beta_1 \dots \beta_{k_2} \in$ Traces Q such that $Q \xrightarrow{\sigma_2} Q'$
As they are strong bis. $\exists P'$ s.t $P \xrightarrow{\sigma_2} P'$,
so $\sigma_2 \in$ Traces (P)

$\Rightarrow$ Traces $(Q) =$ Traces (P).

* As comp. traces are a subset of traces, a comp trace for Q is a trace for P. It is also complete otherwise it contradicts $P \sim Q$.

3.8

Ex 3.3 as counter example:

I) $(P,P) \notin R$
II) $(Q,P) \notin R$ although $(P,Q) \in R$
III) If we consider $R \cup R^{-1}$, it is a strong bisimulation by 3.7 and $(P,P) \notin R \cup R^{-1}$ although $(P,Q) \in R \cup R^{-1}$ and $(Q,P) \in R \cup R^{-1}$

Neither R nor $R \cup R^{-1}$ are the largest str. bis over that L.T.S.

3.10 $K \stackrel{def}{=} P$ ¿ $K \sim P$?

If $P \xrightarrow{\alpha} P'$, by CON $K \xrightarrow{\alpha} P'$, and we know $P' \sim P'$
If $K \xrightarrow{\alpha} K'$, CON requires $P \to K'$, and we know $K' \sim K'$.
And this prop. holds iff $K \sim P$.

MC HOMEW 6

(3.14)

(1) (~) preserved by action prefixing.

As $P \sim Q$, $\exists$ bisimulation R s.t. $(P, Q) \in R$

Consider $R' = \{(\alpha.P, \alpha.Q) \mid \forall \alpha \in Act\} \cup R$

Then, $(\alpha.P, \alpha.Q) \in R'$ $\forall \alpha \in Act$

and:

$\alpha.P \xrightarrow{\alpha} P \Rightarrow \alpha.Q \xrightarrow{\alpha} Q$ and $(P, Q) \in R \subseteq R'$

$\alpha.Q \xrightarrow{\alpha} Q \Rightarrow \alpha.P \xrightarrow{\alpha} P$ and $(P, Q) \in R \subseteq R'$

(2) (+) preserves (~)   As $P \sim Q$, $\exists R$ bisim s.t. $(P, Q) \in R$

Consider $R' = \{(P^* + R^*, Q^* + R^*) \mid \forall R \in Proc\} \cup R$

$(P+R, Q+R) \in R$ $\forall R$ CCS proc. $\cup \{(P,P) : P \text{ proc}\}$ → Is bisim.

Let $(P^* + R', Q^* + R') \in R'$

If $P^* + R' \xrightarrow{\alpha} S$:

(1) $P^* \xrightarrow{\alpha} P''$ and $S = P''$. As $P^* \sim Q^*$, $Q^* \xrightarrow{\alpha} Q''$, so $Q^* + R' \xrightarrow{\alpha} Q''$ and, as $P'' \sim Q''$, $\Rightarrow (P', R') \in R \subseteq R'$

(2) $R^* \xrightarrow{\alpha} R''$ and $S = R''$. $\Rightarrow Q + R \xrightarrow{\alpha} R'$ and $(R', R') \in R$

Similar for $Q + R' \xrightarrow{\alpha} S$:

③ rellabeling preserves $\sim$

Suppose $R_f = \{ (P'[f], Q'[f]) \mid P' \sim Q' \dots \}$

for some f valid rellabeling func.

As $P \sim Q$, $(P[f], Q[f]) \in R$

Let $(P'[f], Q'[f]) \in R$

If $P'[f] \xrightarrow{\alpha} P''[f] \Rightarrow$

$\exists \beta : f(\beta) = \alpha$ and $P' \xrightarrow{\beta} P''$

As $P' \sim Q'$, $Q' \xrightarrow{\beta} Q''$ with $P'' \sim Q''$

so $Q'[f] \xrightarrow{\alpha = f(\beta)} Q''[f]$ and $(P''[f], Q''[f]) \in R$

similar for $Q'[f] \xrightarrow{\alpha} Q''[f]$.

3.17

1. Sym. consider $I = \{ (S, S) : S \in Proc \}$

Let $S \in Proc$, $(S, S) \in I$

and $S \precsim S$ because if α is an action and

$S \xrightarrow{\alpha} S'$, then $S \xrightarrow{\alpha} S'$ and $(S', S') \in I$

Trans.

Let $P, Q, R \in Proc$ s.t. $P \precsim Q^{\Rightarrow \exists T_1}$ and $Q \precsim R^{\Rightarrow \exists T_2}$

consider $T = \{ (A, B) : (A, C) \in T_1 \wedge (C, B) \in T_2$ for some $C \in Proc \}$

$(P, R) \in T$ and $P \precsim R$ because if α is an action.

If $P \xrightarrow{\alpha} P'$ then

$Q \xrightarrow{\alpha} Q'$ and then

$R \xrightarrow{\alpha} R'$ with $(P', Q') \in T_1$ and $(Q', R') \in T_2$

so $(P', R') \in T$.

$\Rightarrow \precsim$ Is a preorder.

It remains to prove $\simeq$ eq. relation.

(3.17)

II

a) $a.0 \sqsubseteq a.a.0$ $R = \{(a.0, a.a.0), (0, a.0)\}$

$a.0 \xrightarrow{a} 0$
$a.a.0 \xrightarrow{a} a.0$

b) $a.b.0 + a.c.0 \sqsubseteq a.(b.0 + c.0)$

$a.b.0 + a.c.0$
$\xrightarrow{a} b.0 \xrightarrow{b} 0$
$\xrightarrow{a} c.0 \xrightarrow{c} 0$

$R = \{(a.b.0 + a.c.0, a.(b.0 + c.0)),$
$(b.0, b.0 + c.0),$
$(c.0, b.0 + c.0),$
$(0, 0)$
$\}$

$a.(b.0 + c.0)$
$\xrightarrow{a} b.0 + c.0 \xrightarrow{b} 0$
$\xrightarrow{c} 0$

a) converse do not hold: Suppose $a.a.0 \; R \; a.0$ simulation.

$(a.a.0, a.0) \in R$

$a.a.0 \xrightarrow{a} a.0$ and $(a.0, 0) \in R$
$a.0 \xrightarrow{a} 0$

$a.0 \xrightarrow{a} 0$ so R is not a simulation.
$0 \not\xrightarrow{a}$

b) converse do not hold: Suppose $a.b.0 + a.c.0 \; R \; a.(b.0 + c.0)$ simulation. Tizga

$(a.b.0 + a.c.0, a.(b.0 + c.0)) \in R$ Tizga

$a.(b.0 + c.0) \xrightarrow{a} b.0 + c.0$
$a.b.0 \xrightarrow{a} b.0$ And $(b.0 + c.0, b.0) \in R$
$a.b.0 \xrightarrow{a} c.0$ or $(b.0 + c.0, c.0) \in R$

$b.0 + c.0 \xrightarrow{c}$ $b.0 + c.0 \xrightarrow{b}$
$b.0 \not\xrightarrow{c}$ $c.0 \not\xrightarrow{b}$

so R is not a simulation.

$\sim \; \subseteq \; \sqsubseteq$

III Let $P \sim Q$ $\exists R$ bisimil. with $(P, Q) \in R$.
If $P \xrightarrow{\alpha} P'$ then $Q \xrightarrow{\alpha} Q'$ with $(P', Q') \in R$, so R is also a simulation and $P \sqsubseteq Q$.

* The converse does not hold: for example in II.a. counterexample.
* A process is always a simulation of itself, for example.
* A process that can simulate any other: $P \overset{def}{=} \sum_{a \in Act} a.P$

But it should be proved.

3.18 Sym

I). $I = \{(Q,Q) : Q \in Proc\}$ Let $P \in Proc$, then $(P,P) \in I$ and

If $P \xrightarrow{\alpha} P'$, then $P \xrightarrow{\alpha} P'$ and $(P',P') \in I$.

If $P \not\xrightarrow{\alpha} P'$, then $P \not\xrightarrow{\alpha}$

Trans.

Let $P \sqsubseteq_{RS} Q$ and $Q \sqsubseteq_{RS} R$. $\exists T_1, T_2$ ready simulations s.t $(P,Q) \in T_1$ and $(Q,R) \in T_2$

Consider $T = \{(A,B) : A T_1 C \wedge C T_2 B$ for some $C \in Proc\}$

$(P,R) \in T$ and

If $P \xrightarrow{\alpha} P'$ then

$Q \xrightarrow{\alpha} Q'$ and

$R \xrightarrow{\alpha} R'$ with $(P',Q') \in T_1$ and $(Q',R') \in T_2$

so $(P',R') \in T$

If $R \not\xrightarrow{\alpha}$ then $Q \not\xrightarrow{\alpha}$ and then $P \not\xrightarrow{\alpha}$.

$\Rightarrow \sqsubseteq_{RS}$ is a preorder.

It remains to prove $\simeq_{RS}$ eq. relation.

II and III) Are similar to 3.17

* In this case $\nexists$ a process which can RS any other.
It should be proved.

3.19 TODO review and think a practical example.

3.20

I) $a.0 \approx a.\tau.0$ $R = \{(a.0, a.\tau.0), (0,0), (\tau.0, 0)\}$

$\begin{cases} a.0 \xrightarrow{a} 0 \\ a.\tau.0 \xRightarrow{a} 0 \end{cases}$

$\begin{cases} a.\tau.0 \xrightarrow{a} \tau.0 \\ a.0 \xRightarrow{a} 0 \end{cases}$ $\begin{cases} \tau.0 \xrightarrow{\tau} 0 \\ 0 \xRightarrow{\hat{\tau}} 0 \end{cases}$

II) SmUni ~~[illegible]~~ $\not\approx$ Start   Suppose SmUni R Start with R weak bisim.

$(SmUni, Start) \in R$

$Start \xRightarrow{pub} \underbrace{(CM_b \mid CS_1) \setminus \{coin, coffee\}}_{B}$) $\in R$

$SmUni \xRightarrow{pub} A$

any reachable of SmUni.

Now, $A \xrightarrow{\tau}$
$B \xRightarrow{\tau}$
at some point $A \xrightarrow{pub}$ and $B \not\xRightarrow{pub}$

* SmUni $\approx$ Spec is easy to do.

MC. Homework 7

3.21 2) TODO

$L = \{\, p, v \,\}$

3) $FMutex \approx Mutex_2$

$$\begin{cases} FMutex \xrightarrow{\tau} ((enter.exit.\bar{v}.User \mid \bar{v}.Sem) \mid FUser) \setminus L & \xrightarrow{enter.\,exit.\,\tau}{}^{*} \text{ Initial point.} \\ Mutex_2 \overset{\tau}{\Rightarrow} ((enter.exit.\bar{v}.User \mid \bar{v}.Sem) \mid User) \setminus L \end{cases}$$

$$\begin{cases} FMutex \xrightarrow{\tau} ((User \mid \bar{v}.Sem) \mid enter.(exit.\bar{v}.FUser + exit.\bar{v}.0)) \setminus L \\ Mutex_2 \overset{\tau}{\Rightarrow} ((User \mid \bar{v}.Sem) \mid enter.exit.\bar{v}.User) \setminus L \end{cases}$$

. enter
. exit of left branch, we reach initial point.

$Mutex \xrightarrow{\tau}$ match both with $\overset{\tau}{\Rightarrow}$ of FMutex of the left User. $\rightarrow^{*}$ Initial point.

The interesting part:

$$(\underbrace{((User \mid \bar{v}.Sem) \mid \bar{v}.0) \setminus L}_{A},\ \underbrace{((User \mid \bar{v}.Sem) \mid \bar{v}.User) \setminus L}_{B})$$

$A \xrightarrow{\tau} ((User \mid Sem) \mid 0) \setminus L$

$B \xrightarrow{\tau} ((User \mid Sem) \mid User) \setminus L$

At this point, continue matching B actions with the left User and it will work.

3.23 Is similar to the proof for strong bisim.

3.24 $\sim \subseteq \approx$ also easy.

3.25 ... 3.28 left todo and also 3.29 - 3.31

3.26 Properties of $\approx$ I have left to do, also for $\sim$ previously.

3.29 Let $P \in Proc$. $P \setminus (Act - \{\tau\}) \approx 0$

If $P \setminus (Act - \{\tau\}) \stackrel{\tau}{\not\to}$ trivial.

If $P \setminus (Act - \{\tau\}) \xrightarrow{\tau} P' \setminus \underbrace{(Act - \{\tau\})}_{\searrow \text{ via the SOS rules.}}$

and $0 \stackrel{\tau}{\Rightarrow} 0$

So we can consider $R = \{(P \setminus (Act - \{\tau\}), 0) : P \in Proc\}$ and it is a weak bisimulation.

* This does not hold for $\sim$. Counterexample: $\tau.0 \setminus (Act - \{\tau\}) \not\sim 0$.

$\approx$ Is not a congruence

$0 \approx \tau.0$ but $a.0 + 0 \not\approx a.0 + \tau.0$

$\begin{cases} a.0 + 0 \xrightarrow{a} 0 \\ a.0 + \tau.0 \stackrel{a}{\Rightarrow} 0 \end{cases}$ $\begin{cases} a.0 + \tau.0 \xrightarrow{\tau} 0 \\ a.0 + 0 \stackrel{\tau}{\Rightarrow} a.0 + 0 \end{cases}$ $\rightarrowtail$ $a.0 + 0 \xrightarrow{a}$, $0 \stackrel{a}{\not\Rightarrow}$

Theorem 3.4 for remaining ops.

3.33 Give a restriction over CCS syntax for $\approx$ to be a cong.

3.36 $\approx^c$ is an equivalence that is a congruence.

3.37 $S \not\approx T$:

Attacker: $S \xrightarrow{a} S_1$ $\Rightarrow$ Defender $T \xrightarrow{a} T_1$

$T_1 \xrightarrow{b} T_1$ $\Rightarrow$ $S_1 \xrightarrow{b} S_2$

~~[illegible]~~ $\Rightarrow$ $T_1 \stackrel{a}{\not\to}$

$S_2 \xrightarrow{a} S$

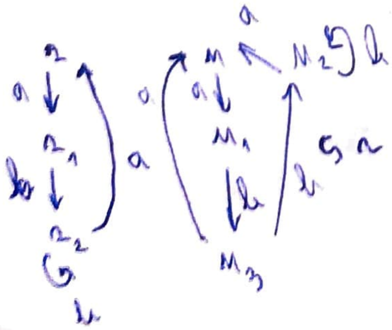
$S \not\approx V$:

Attacker: $S \xrightarrow{a} S_1$ $\Rightarrow$ Defender $V \xrightarrow{a} V_1$

~~[illegible]~~ $\Rightarrow$ $S_1 \xrightarrow{b} S_2$

$V_1 \xrightarrow{b} V_3$

$S_2 \xrightarrow{a} S$ $\Rightarrow$ $V_3 \stackrel{a}{\not\to}$



$S \approx M$: Iff $S \xrightarrow{a} S_1$, $M \xrightarrow{a} M_1$ $\Rightarrow$ Iff $S_1 \xrightarrow{b} S_2$, $M_1 \xrightarrow{b} M_3$ $\Rightarrow$ Iff $S_2 \xrightarrow{b} S_2$, $M_3 \xrightarrow{b} M_2$ $\Rightarrow$ Iff $S_2 \xrightarrow{b} S_2$, $M_2 \xrightarrow{b} M_2$

Iff $S_2 \xrightarrow{a} S$, $M_3 \to M$ Iff $S_2 \xrightarrow{a} S$, $M_2 \xrightarrow{a} M$

M.C. Hansen B.

5.1

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

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

5.2

$CS \overset{def}{=} \overline{pub}.\overline{coin}.coffee.CS$

I) $[coffee]\langle biscuit\rangle tt$

$[\![tt]\!] = Proc$

$[\![\langle biscuit\rangle Proc]\!] = \langle\cdot biscuit\cdot\rangle Proc =$

$= \{P \mid P \xrightarrow{biscuit}\} = \emptyset$ in the case of CS

$[\![[coffee]\emptyset]\!] = [\cdot coffee\cdot]\emptyset = \{P \mid P \not\xrightarrow{coffee}\}$

$CS \in [coffee]\langle biscuit\rangle tt$ $\quad (CS \models [coffee]\langle biscuit\rangle tt)$

But this is not interesting because she cannot drink a coffee now.

II) $\langle coffee\rangle tt \wedge \langle tea\rangle tt$

$\langle coffee\rangle tt \wedge [tea]ff$

$[coffee][coffee]\langle tea\rangle tt$

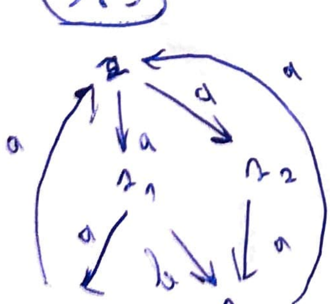
III) $\langle a\rangle ff$ $\quad \langle\cdot a\cdot\rangle\emptyset = \{P \xrightarrow{a} P'$ for some $P' \in \emptyset\}$

$= \emptyset$

$[a]tt$ $\quad [\cdot a\cdot]Proc = \{P : \forall P \xrightarrow{a} P'$ then $P' \in Proc\}$

$= Proc$

5.3



I)

$s \models \langle a\rangle tt$

$s \not\models \langle b\rangle tt$

$s \not\models [a]ff$

$s \models [b]ff$

$s \not\models [a]\langle b\rangle tt$

$s \models \langle a\rangle\langle b\rangle tt$

$s \models [a]\langle a\rangle[a][b]ff$

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

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

$s \not\models \langle a\rangle([b][a]ff \wedge \langle b\rangle tt)$

$s \not\models \langle a\rangle([a](\langle a\rangle tt \wedge \boxed{[b]}ff) \wedge \underbrace{\langle b\rangle ff}_{\emptyset})$

II) $[\![[a][b]ff]\!]$

$[\![ff]\!] = \emptyset$

$[\![[b]\emptyset]\!] = \{ P : P \not\xrightarrow{b} \}$

$[\![[a][b]ff]\!] = \{ P : P \xrightarrow{a} Q \Rightarrow Q \not\xrightarrow{b} \}$

$[\![\langle a\rangle (\langle a\rangle tt \wedge \langle b\rangle tt]\!] = \{ P : P \xrightarrow{aa} \wedge P \xrightarrow{ab} \}$

$[\![\langle a\rangle tt]\!] = \{ P : P \xrightarrow{a} \}$

$[\![\langle b\rangle tt]\!] = \{ P : P \xrightarrow{b} \}$

$[\![[a][a][b]ff]\!] = \{ P : P \xrightarrow{a} P' \Rightarrow P' \xrightarrow{a} P'' \Rightarrow P'' \not\xrightarrow{b} \}$

$[\![[a](\langle a\rangle tt \vee \langle b\rangle tt)]\!] = \{ P : P \xrightarrow{a} P' \Rightarrow P' \xrightarrow{a} \vee P' \xrightarrow{b} \}$

5.4

I) $[\![[tock]ff]\!] = \{ Clock \}$. because $Clock \xrightarrow{tick} Clock$.

$[\![\langle tick\rangle tt]\!] = \{ Clock \}$ because $Clock \xrightarrow{tick} Clock$

$[\![[tick]\{Clock\}]\!] = \{ Clock \}$

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

II) 0 times, $Clock \models tt$ Induction.

If $Clock \models \underbrace{\langle tick\rangle \ldots \langle tick\rangle}_{n} tt$ ~~...~~

$\Leftarrow$

$Clock \models \underbrace{\langle tick\rangle \ldots \langle tick\rangle}_{n+1} tt$.

$Clock \in [\![\underbrace{\langle tick\rangle \ldots \langle tick\rangle}_{n} tt]\!] = S$

$[\![\langle tick\rangle S]\!] = \{ P : P \xrightarrow{tick} Q$ for some $Q \in S \}$

$Clock \in$ because $Clock \xrightarrow{tick} Clock$

5.5 $F = [a](\langle b\rangle tt \wedge \langle c\rangle tt)$

$a.b.0 + a.c.0 \not\models F$

$a.(b.0 + c.0) \models F$

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

$\Downarrow$

$G = \cancel{[a][b](\langle c\rangle tt \vee \langle d\rangle tt)}$

$a.(b.c.0 + b.d.0) \models G$

$a.b.c.0 + a.b.d.0 \not\models G$

M.C. Hansen 9.

(5.6)

If $F = tt$, $P \models tt \leftrightarrow P \in [\![tt]\!] = Proc$
$\forall P \in Proc$ $\quad \forall P \in Proc$

If $F = ff$, $P \not\models ff \leftrightarrow P \notin [\![ff]\!] = \emptyset$
$\forall P \in Proc$ $\quad \forall P \in Proc$

If $F = F_1 \wedge F_2$. Let $P \in Proc$

If $P \models F_1$ and $P \models F_2$, by ind. hyp
$P \in [\![F_1]\!]$ and $P \in [\![F_2]\!]$, so $P \in [\![F_1 \wedge F_2]\!]$

If $P \in [\![F_1 \wedge F_2]\!]$ then $P \in [\![F_1]\!] \cap [\![F_2]\!]$.
By ind. hyp. $P \models F_1$ and $P \models F_2$, so $P \models F_1 \wedge F_2$

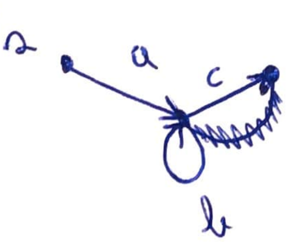
If $F = F_1 \vee F_2$. Similar

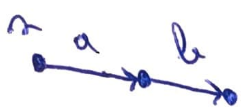
If $F = \langle a \rangle G$. Let $P \in Proc$.

If $P \models F$ then $P \xrightarrow{a} P'$ for some P' s.t. $P' \models G$
By ind. hyp., $P' \in [\![G]\!]$
so $P \in \langle \cdot a \cdot \rangle [\![G]\!]$ and then $P \in [\![\langle a \rangle G]\!]$

If $P \in [\![\langle a \rangle G]\!]$, then $\exists P'$ s.t. $P \xrightarrow{a} P'$ and $P' \in [\![G]\!]$
By ind. hyp. $P' \models G$, so
$P \models \langle a \rangle G$

If $F = [a] G$, similar.

(5.7) I) 

II) 

III) 

~~Thes~~

(Prop. 5.1) Proof TODO

(Exercise 5.8)

Theorem 5.1

Ex 5.9 Let P, Q be image finite processes.

$\Rightarrow$ If $P \sim Q$. Let $P \models F$ ~~iff~~ for a formula F.

If $F = tt$. $P \in Proc$ and $Q \in Proc$, so $Q \models F$

If $F = ff$. similar.

If $F = F_1 \wedge F_2$. $P \in [\![F_1]\!]$ and $P \in [\![F_2]\!]$

by ind. hyp, $Q \models F_1 \wedge Q \models F_2$, so $Q \models F$

If $F = F_1 \vee F_2$. similar.

If $F = [a]G$

This branch is not necessary.

If $P \xrightarrow{a} P'$ then $P' \models G$.
$Q \xrightarrow{a} Q'$ and $P' \sim Q'$. By ind hyp. $Q' \models G$

If $Q \xrightarrow{a} Q'$
$P \xrightarrow{a} P'$ and $P' \sim Q'$. Also $P' \models G$.
By ind. h. $Q' \models G$

so $Q \models F$.

F or $\langle a \rangle G$ similar.

By symmetry of $\sim$, If $Q \models F \Rightarrow P \models F$.

$\Leftarrow$

Why not $R' \not\models F_i$ and $S_i \models F_i$ for some i?
To simplify, we would consider F_i^c in that case.

If $n = 0$, the formula is $\langle a \rangle\, tt$

R can perform a
but S cannot perform a.