Advanced Process Calculi

Lecture 3: bisimulation in psi-calculi

Copenhagen, August 2013

Joachim Parrow

Psi-calculi

T	(Data) Terms	M, N
A	Assertions	Ψ, Ψ'
C	Conditions	φ, φ'

Channel Object Pattern Test (aka guard)

$\overline{M}\, N.P$	Output
$\underline{M}(\lambda\widetilde{x})N.P$	Input
$\mathbf{case}\ \varphi_1 : P_1 \,[\,]\, \cdots \,[\,]\, \varphi_n : P_n$	Case
$(\nu a)P$	Restriction
$P \mid Q$	Parallel
$!P$	Replication
$(\!\vert \Psi \vert\!)$	Assertion

To mean tell Ψ

Instance parameters

$$\boxed{X[\widetilde{x} := \widetilde{T}]}$$

Equivariance: $p \cdot (X[\tilde{x} := \tilde{T}]) = (p \cdot X)\big[(p \cdot \tilde{x}) := (p \cdot \tilde{T})\big]$

Freshness: if $\tilde{x} \subseteq \mathrm{n}(X)$ and $a \# X[\tilde{x} := \tilde{T}]$ then $a \# \tilde{T}$

Alpha-equivalence: if $p \subseteq \tilde{x} \times (p \cdot \tilde{x})$ and $(p \cdot \tilde{x}) \# X$ then
$X[\tilde{x} := \tilde{T}] = (p \cdot X)[(p \cdot \tilde{x}) := \tilde{T}]$

$$\begin{array}{l} \dot{\leftrightarrow} : \mathbf{T} \times \mathbf{T} \to \mathbf{C} \\ \otimes : \mathbf{A} \times \mathbf{A} \to \mathbf{A} \\ \mathbf{1} : \mathbf{A} \\ \vdash \subseteq \mathbf{A} \times \mathbf{C} \end{array}$$

Channel Symmetry: $\Psi \vdash M \dot{\leftrightarrow} N \implies \Psi \vdash N \dot{\leftrightarrow} M$

Channel Transitivity: $\Psi \vdash M \dot{\leftrightarrow} N \ \wedge \ \Psi \vdash N \dot{\leftrightarrow} L$
$\implies \Psi \vdash M \dot{\leftrightarrow} L$

Composition: $\Psi \simeq \Psi' \implies \Psi \otimes \Psi'' \simeq \Psi' \otimes \Psi''$

Identity: $\Psi \otimes \mathbf{1} \simeq \Psi$

Associativity: $(\Psi \otimes \Psi') \otimes \Psi'' \simeq \Psi \otimes (\Psi' \otimes \Psi'')$

Commutativity: $\Psi \otimes \Psi' \simeq \Psi' \otimes \Psi$

All the rules

$$\textsc{In}\ \frac{\Psi \vdash M \dot{\leftrightarrow} K}{\Psi \rhd \underline{M}(\lambda\widetilde{y})N.P \xrightarrow{\underline{K}\,N[\widetilde{y}:=\widetilde{L}]} P[\widetilde{y}:=\widetilde{L}]} \qquad \textsc{Out}\ \frac{\Psi \vdash M \dot{\leftrightarrow} K}{\Psi \rhd \overline{M}\,N.P \xrightarrow{\overline{K}\,N} P} \qquad \textsc{Case}\ \frac{\Psi \rhd P_i \xrightarrow{\alpha} P' \qquad \Psi \vdash \varphi_i}{\Psi \rhd \mathbf{case}\ \widetilde{\varphi} : \widetilde{P} \xrightarrow{\alpha} P'}$$

$$\textsc{Com}\ \frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow{\overline{M}\,(\nu\widetilde{a})N} P' \qquad \Psi_P \otimes \Psi \rhd Q \xrightarrow{\underline{K}\,N} Q' \qquad \Psi \otimes \Psi_P \otimes \Psi_Q \vdash M \dot{\leftrightarrow} K}{\Psi \rhd P \mid Q \xrightarrow{\tau} (\nu\widetilde{a})(P' \mid Q')}\ \widetilde{a}\#Q$$

$$\textsc{Par}\ \frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow{\alpha} P'}{\Psi \rhd P|Q \xrightarrow{\alpha} P'|Q}\ \mathrm{bn}(\alpha)\#Q \qquad \textsc{Scope}\ \frac{\Psi \rhd P \xrightarrow{\alpha} P'}{\Psi \rhd (\nu b)P \xrightarrow{\alpha} (\nu b)P'}\ b\#\alpha,\Psi$$

$$\textsc{Open}\ \frac{\Psi \rhd P \xrightarrow{\overline{M}\,(\nu\widetilde{a})N} P'}{\Psi \rhd (\nu b)P \xrightarrow{\overline{M}\,(\nu\widetilde{a}\cup\{b\})N} P'}\ \begin{array}{c} b\#\widetilde{a},\Psi,M \\ b \in \mathrm{n}(N) \end{array} \qquad \textsc{Rep}\ \frac{\Psi \rhd P \mid !P \xrightarrow{\alpha} P'}{\Psi \rhd !P \xrightarrow{\alpha} P'}$$

+ freshness conditions in Par and Com

+ symmetric variants

Correctness

How can we prove **beyond doubt** that we have made no mistake similar to applied pi and CC-pi?

Correctness

How can we prove **beyond doubt** that we have made no mistake similar to applied pi and CC-pi?

I.e., prove that if P and Q have the same transitions then so do $P|R$ and $Q|R$?

Correctness

How can we prove **beyond doubt** that we have made no mistake similar to applied pi and CC-pi?

I.e., prove that if P and Q have the same transitions then so do $P|R$ and $Q|R$?

...and that it satisfies the scoping laws, eg scope extension, that $(\nu a)P \mid Q$ and $(\nu a)(P|Q)$ have the same transitions if $a\#Q$

Strategy

Define an intuitive equivalence from the semantics

Prove that it is a congruence

Prove that it is satisfies universal laws like scope extension

ie exactly the same strategy as in the pi-calculus

Different instances may have different laws!

Different instances may have different laws!

$\mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\, P$ $\qquad$ $\mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\, \mathbf{if}\ \varphi\ \mathbf{then}\ P$

Different instances may have different laws!

$\textbf{if}\ \varphi\ \textbf{then}\ \tau.\, P$ $\qquad$ $\textbf{if}\ \varphi\ \textbf{then}\ \tau.\, \textbf{if}\ \varphi\ \textbf{then}\ P$

only has a transition in an environment Ψ
such that $\Psi \vdash \varphi$

Different instances may have different laws!

$\textbf{if } \varphi \textbf{ then } \tau.\, P$ $\qquad\qquad$ $\textbf{if } \varphi \textbf{ then } \tau.\, \textbf{if } \varphi \textbf{ then } P$

only has a transition in an environment Ψ such that $\Psi \vdash \varphi$

lead to P $\qquad\qquad$ $\textbf{if } \varphi \textbf{ then } .\, P$

Different instances may have different laws!

$\mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,P$ $\qquad\qquad$ $\mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,\mathbf{if}\ \varphi\ \mathbf{then}\ P$

only has a transition in an environment Ψ
such that $\Psi \vdash \varphi$

lead to P $\qquad\qquad$ $\mathbf{if}\ \varphi\ \mathbf{then}\ .\,P$

The environment used to satisfy φ
Will it always do this?
Can it change to make φ false?

The environment is any agent, and this can evolve!

Eg: $Q = (\!|\Psi|\!) \mid \tau \,.\, (\!|\Psi'|\!)$

$\mathcal{F}(Q) = \Psi$

The environment is any agent, and this can evolve!

Eg: $Q = (\!|\Psi|\!) \mid \tau\,.\,(\!|\Psi'|\!)$

$\mathcal{F}(Q) = \Psi$

Q can evolve to Q' with $\mathcal{F}(Q') = \Psi \otimes \Psi'$

The environment is any agent, and this can evolve!

Eg: $Q = (\!|\Psi|\!) \mid \tau \, . \, (\!|\Psi'|\!)$

$\mathcal{F}(Q) = \Psi$

Q can evolve to Q' with $\mathcal{F}(Q') = \Psi \otimes \Psi'$

$Q \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\, P$ $\qquad$ $Q \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\, \mathbf{if}\ \varphi\ \mathbf{then}\ P$

The environment is any agent, and this can evolve!

Eg: $Q = (\!|\Psi|\!) \mid \tau\,.\,(\!|\Psi'|\!)$

$\mathcal{F}(Q) = \Psi$

Q can evolve to Q' with $\mathcal{F}(Q') = \Psi \bigotimes \Psi'$

$Q \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,P$ $\qquad$ $Q \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,\mathbf{if}\ \varphi\ \mathbf{then}\ P$

evolve to

$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P$ $\qquad$ $(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$

$$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P \qquad\qquad (\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

$$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P \qquad\qquad (\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

Do these behave similarly? We know that $\Psi \vdash \varphi$

$$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P \qquad\qquad (\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

Do these behave similarly? We know that $\Psi \vdash \varphi$

Do we then also know that $\Psi \otimes \Psi' \vdash \varphi$?

If so the agents behave the same

$$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P \qquad\qquad (\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

Do these behave similarly? We know that $\Psi \vdash \varphi$

Do we then also know that $\Psi \otimes \Psi' \vdash \varphi$?

If so the agents behave the same

If not they are different since only the left can act as P

$$(\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid P \qquad\qquad (\!|\Psi|\!) \mid (\!|\Psi'|\!) \mid \ \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

Do these behave similarly? We know that $\Psi \vdash \varphi$

Do we then also know that $\Psi \otimes \Psi' \vdash \varphi$?

If so the agents behave the same

If not they are different since only the left can act as P

The crucial question is if the psi-calculus satisfies

$$\Psi \vdash \varphi \ \Rightarrow \Psi \otimes \Psi' \vdash \varphi$$ **monotonicity**

The only thing we know for sure about any psi-calculus:

$$\boxed{X[\widetilde{x} := \widetilde{T}]}$$

Equivariance:	$p \cdot (X[\tilde{x} := \tilde{T}]) = (p \cdot X)\big[(p \cdot \tilde{x}) := (p \cdot \tilde{T})\big]$
Freshness:	if $\tilde{x} \subseteq \mathrm{n}(X)$ and $a \# X[\tilde{x} := \tilde{T}]$ then $a \# \tilde{T}$
Alpha-equivalence:	if $p \subseteq \tilde{x} \times (p \cdot \tilde{x})$ and $(p \cdot \tilde{x}) \# X$ then $X[\tilde{x} := \tilde{T}] = (p \cdot X)[(p \cdot \tilde{x}) := \tilde{T}]$

$$\boxed{\begin{array}{l} \dot{\leftrightarrow} : \mathbf{T} \times \mathbf{T} \to \mathbf{C} \\ \otimes \ : \mathbf{A} \times \mathbf{A} \to \mathbf{A} \\ \mathbf{1} \ : \mathbf{A} \\ \vdash \subseteq \mathbf{A} \times \mathbf{C} \end{array}}$$

Channel Symmetry:	$\Psi \vdash M \dot{\leftrightarrow} N \implies \Psi \vdash N \dot{\leftrightarrow} M$
Channel Transitivity:	$\Psi \vdash M \dot{\leftrightarrow} N \ \wedge \ \Psi \vdash N \dot{\leftrightarrow} L \implies \Psi \vdash M \dot{\leftrightarrow} L$

Composition:	$\Psi \simeq \Psi' \implies \Psi \otimes \Psi'' \simeq \Psi' \otimes \Psi''$
Identity:	$\Psi \otimes \mathbf{1} \simeq \Psi$
Associativity:	$(\Psi \otimes \Psi') \otimes \Psi'' \simeq \Psi \otimes (\Psi' \otimes \Psi'')$
Commutativity:	$\Psi \otimes \Psi' \simeq \Psi' \otimes \Psi$

It turns out this does not imply monotonicity!

So there are psi-calculi where this does not hold

It turns out this does not imply monotonicity!

So there are psi-calculi where this does not hold

Good: this means we can have psi-calculi describing concurrent constraints with retracts!

It turns out this does not imply monotonicity!

So there are psi-calculi where this does not hold

Good: this means we can have psi-calculi describing concurrent constraints with retracts!

Bad: this means that some natural looking laws are not valid in these calculi

$$\mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,P \quad \overset{?}{\sim} \quad \mathbf{if}\ \varphi\ \mathbf{then}\ \tau.\,\mathbf{if}\ \varphi\ \mathbf{then}\ P$$

$$\textbf{if } \varphi \textbf{ then } \tau.\, P \quad \overset{?}{\sim} \quad \textbf{if } \varphi \textbf{ then } \tau.\, \textbf{if } \varphi \textbf{ then } P$$

This law is only valid if assertion composition is monotonic

$$\textbf{if } \varphi \textbf{ then } \tau . P \quad \stackrel{?}{\sim} \quad \textbf{if } \varphi \textbf{ then } \tau . \textbf{if } \varphi \textbf{ then } P$$

This law is only valid if assertion composition is monotonic

In fact this is expected: if the environment can evolve to retract facts the law should not be valid!

Universal laws

Universal laws are those valid in **all** psi-calculi

We should at least expect something like

$$
\begin{array}{rcll}
P & \sim & P \mid \mathbf{0} & \\
P \mid (Q \mid R) & \sim & (P \mid Q) \mid R & \\
P \mid Q & \sim & Q \mid P & \\
(\nu a)\mathbf{0} & \sim & \mathbf{0} & \\
P \mid (\nu a)Q & \sim & (\nu a)(P \mid Q) & \text{if } a \# P \\
\overline{M}\ N.(\nu a)P & \sim & (\nu a)\overline{M}\ N.P & \text{if } a \# M, N \\
\underline{M}(\lambda \widetilde{x})N.(\nu a)P & \sim & (\nu a)\underline{M}(\lambda \widetilde{x})(N).P & \text{if } a \# M, N \\
\textbf{case}\ \widetilde{\varphi} : \widetilde{(\nu a)P} & \sim & (\nu a)\textbf{case}\ \widetilde{\varphi} : \widetilde{P} & \text{if } a \# \widetilde{\varphi} \\
(\nu a)(\nu b)P & \sim & (\nu b)(\nu a)P & \\
!P & \sim & P \mid !P &
\end{array}
$$

Universal laws

Universal laws are those valid in **all** psi-calculi

If any of thee fail something is wrong with our definitions!

$$
\begin{array}{rcll}
P & \sim & P \mid \mathbf{0} & \\
P \mid (Q \mid R) & \sim & (P \mid Q) \mid R & \\
P \mid Q & \sim & Q \mid P & \\
(\nu a)\mathbf{0} & \sim & \mathbf{0} & \\
P \mid (\nu a)Q & \sim & (\nu a)(P \mid Q) & \text{if } a \# P \\
\overline{M}\ N.(\nu a)P & \sim & (\nu a)\overline{M}\ N.P & \text{if } a \# M, N \\
\underline{M}(\lambda \widetilde{x})N.(\nu a)P & \sim & (\nu a)\underline{M}(\lambda \widetilde{x})(N).P & \text{if } a \# M, N \\
\textbf{case}\ \widetilde{\varphi} : \widetilde{(\nu a)P} & \sim & (\nu a)\textbf{case}\ \widetilde{\varphi} : \widetilde{P} & \text{if } a \# \widetilde{\varphi} \\
(\nu a)(\nu b)P & \sim & (\nu b)(\nu a)P & \\
!P & \sim & P \mid !P &
\end{array}
$$

Universal laws

We also want compositionality

Ie the equivalence is a congruence

Bisimulation

First attempt, can we as in the pi-calculus define bisimulation only in terms of transitions?

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P\,R\,Q$ implies that $\forall\alpha.\mathrm{bn}(\alpha)\#Q$.
   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P'\,R\,Q'$

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P \, R \, Q$ implies that $\forall \alpha . \mathrm{bn}(\alpha) \# Q$.

   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P\,R\,Q$ implies that $\forall\alpha.\mathrm{bn}(\alpha)\#Q$.

   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P'\,R\,Q'$

Bad News:

This is clearly inadequate since also the frames of P and Q must be related

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P\,R\,Q$ implies that $\forall\alpha.\mathrm{bn}(\alpha)\#Q$.
   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P'\,R\,Q'$

Bad News:

This is clearly inadequate since also the frames of P and Q must be related

This would equate all agents of form $(\!|\Psi|\!)$ since these have no transitions!

Bad News:

This is clearly inadequate since also the frames of P and Q must be related

This would equate all agents of form $(\!|\Psi|\!)$ since these have no transitions!

Remedy(?) Also take account of the frames

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P \, R \, Q$ implies that $\forall \alpha . \mathrm{bn}(\alpha) \# Q$.

   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$

3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

Remedy(?) Also take account of the frames

The binary relation R is a *bisimulation* if

1. R is symmetric
2. $P\,R\,Q$ implies that $\forall\alpha.\mathrm{bn}(\alpha)\#Q$.
   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P'\,R\,Q'$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

"Can be alpha-converted to each other"

$\boxed{F \vdash \varphi}$ means $F =_\alpha (\nu\widetilde{b})\Psi$

$\widetilde{b}\#\varphi$

$\Psi \vdash \varphi$

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P\,R\,Q$ implies that $\forall\alpha.\mathrm{bn}(\alpha)\#Q$.
   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P'\,R\,Q'$

3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

The binary relation R is a *bisimulation* if

1. R is symmetric
2. $P \mathrel{R} Q$ implies that $\forall \alpha . \mathrm{bn}(\alpha) \# Q$.
   $P \xrightarrow{\alpha} P'$ implies $Q \xrightarrow{\alpha} Q'$ and $P' \mathrel{R} Q'$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

Bad News:

This is not even well formed.
Transitions depend on environment.
Must require this for all possible environments?

Bad News:

This is not even well formed.
Transitions depend on environment.
Must require this for all possible environments?

The binary relation R is a *bisimulation* if

1. R is symmetric

2. $P \mathrel{R} Q$ implies that $\forall \alpha, \Psi . \mathrm{bn}(\alpha) \# Q, \Psi .$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $P' \mathrel{R} Q'$

3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

The binary relation R is a *bisimulation* if

1. R is symmetric
2. $P \, R \, Q$ implies that $\forall \alpha, \Psi . \mathrm{bn}(\alpha) \# Q, \Psi .$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

Actually this is a bit too strong: requires derivatives to bisimulate for all environments even though some may not be reachable!

The binary relation R is a *bisimulation* if

1. R is symmetric
2. $P \, R \, Q$ implies that $\forall \alpha, \Psi . \mathrm{bn}(\alpha) \# Q, \Psi .$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

Actually this is a bit too strong: requires derivatives to bisimulate for all environments even though some may not be reachable!

$$\textbf{if } \varphi \textbf{ then } \tau . P \quad \overset{?}{\sim} \quad \textbf{if } \varphi \textbf{ then } \tau . \textbf{if } \varphi \textbf{ then } P$$

The binary relation R is a *bisimulation* if

1. R is symmetric
2. $P \, R \, Q$ implies that $\forall \alpha, \Psi . \mathrm{bn}(\alpha) \# Q, \Psi .$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $P' \, R \, Q'$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$

Actually this is a bit too strong: requires derivatives to bisimulate for all environments even though some may not be reachable!

$$\mathbf{if}\ \varphi\ \mathbf{then}\ \tau . P \quad \overset{?}{\sim} \quad \mathbf{if}\ \varphi\ \mathbf{then}\ \tau . \mathbf{if}\ \varphi\ \mathbf{then}\ P$$

This will not hold in any psi-calculus
Not even in monotonic ones!

Better: A bisimulation is a **ternary** relation, relating an assertion and two agents.

$R(\Psi, P, Q)$ means that P and Q are bisimilar if the environment of both is Ψ

Better: A bisimulation is a **ternary** relation, relating an assertion and two agents.

$R(\Psi, P, Q)$ means that P and Q are bisimilar if the environment of both is Ψ

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

Better: A bisimulation is a **ternary** relation, relating an assertion and two agents.

$R(\Psi, P, Q)$ means that P and Q are bisimilar if the environment of both is Ψ

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

$P \stackrel{\bullet}{\sim} Q$ if for some bisimulation R it holds $\forall \Psi . R(\Psi, P, Q)$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$

2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$

3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

We are almost there, but not quite!

This definition will not satisfy compositionality!

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\,\mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$

$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$

$$T = \tau\ .\ (\!|\Psi|\!)$$

$$\Psi \vdash \varphi$$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha. \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\ .\ (\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall\alpha.\mathrm{bn}(\alpha)\#Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

$$\begin{aligned} R = \{ \quad & (\Psi, P, Q), \quad (\mathbf{1}, P, Q), \\ & (\Psi, \beta\ .\ \mathbf{0}, \beta\ .\ \mathbf{0}), \quad (\mathbf{1}, \beta\ .\ \mathbf{0}, \beta\ .\ \mathbf{0}), \\ & (\Psi, \mathbf{0}, \mathbf{0}), \quad (\mathbf{1}, \mathbf{0}, \mathbf{0}), \\ & (\Psi,\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\ .\ \mathbf{0},\ \beta\ .\ \mathbf{0}), \\ & (\mathbf{1},\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\ .\ \mathbf{0},\ \mathbf{0})\} \end{aligned}$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \textbf{if}\ \varphi\ \textbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha.\mathrm{bn}(\alpha)\#Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

$$\begin{aligned} R = \{ \quad & (\Psi, P, Q), \quad (\mathbf{1}, P, Q), \\ & (\Psi, \beta\,.\,\mathbf{0}, \beta\,.\,\mathbf{0}), \quad (\mathbf{1}, \beta\,.\,\mathbf{0}, \beta\,.\,\mathbf{0}), \\ & (\Psi, \mathbf{0}, \mathbf{0}), \quad (\mathbf{1}, \mathbf{0}, \mathbf{0}), \\ & (\Psi,\ \textbf{if}\ \varphi\ \textbf{then}\ \beta\,.\,\mathbf{0},\ \beta\,.\,\mathbf{0}), \\ & (\mathbf{1},\ \textbf{if}\ \varphi\ \textbf{then}\ \beta\,.\,\mathbf{0},\ \mathbf{0})\} \end{aligned}$$

Proves $P \sim Q$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \textbf{if}\ \varphi\ \textbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\ .\ (\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \triangleright T \mid P \xrightarrow{\beta} T \mid \textbf{if}\ \varphi\ \textbf{then}\ \beta\ .\ \mathbf{0} \xrightarrow{\tau} (\!|\Psi|\!) \mid \textbf{if}\ \varphi\ \textbf{then}\ \beta\ .\ \mathbf{0}$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \rhd T \mid P \xrightarrow{\beta} T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid ??$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \rhd T \mid P \xrightarrow{\beta} T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid \beta\,.\,\mathbf{0}$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\ .\ (\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \rhd T \mid P \xrightarrow{\beta} \boxed{T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\ .\ \mathbf{0}} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\ .\ \mathbf{0}$$
$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} \boxed{T \mid \beta\ .\ \mathbf{0}}$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \rhd T \mid P \xrightarrow{\beta} T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid \beta\,.\,\mathbf{0}$$
No, since $T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$ has no action β

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \triangleright T \mid P \xrightarrow{\beta} \boxed{T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} \boxed{T \mid \beta\,.\,\mathbf{0}}$$
No, since $T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$ has no action β

$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} T \mid \mathbf{0}$$

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \triangleright T \mid P \xrightarrow{\beta} \boxed{T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} \boxed{T \mid \beta\,.\,\mathbf{0}}$$

No, since $T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$ has no action β

$$\mathbf{1} \triangleright T \mid Q \xrightarrow{\beta} \boxed{T \mid \mathbf{0}}$$

No, since $T \mid \mathbf{0}$ has no action β

$$P = \beta.\beta.\mathbf{0} + \beta.\mathbf{0} + \beta.\ \mathbf{if}\ \varphi\ \mathbf{then}\ \beta.\mathbf{0}$$
$$Q = \beta.\beta.\mathbf{0} + \beta.\mathbf{0}$$
$$T = \tau\,.\,(\!|\Psi|\!)$$
$$\Psi \vdash \varphi$$

$$\mathbf{1} \rhd T \mid P \xrightarrow{\beta} \boxed{T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}} \xrightarrow{\tau} (\!|\Psi|\!) \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$$
$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} T \mid ??$$

$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} \boxed{T \mid \beta\,.\,\mathbf{0}}$$

No, since $T \mid \mathbf{if}\ \varphi\ \mathbf{then}\ \beta\,.\,\mathbf{0}$ has no action β

$$\mathbf{1} \rhd T \mid Q \xrightarrow{\beta} \boxed{T \mid \mathbf{0}}$$

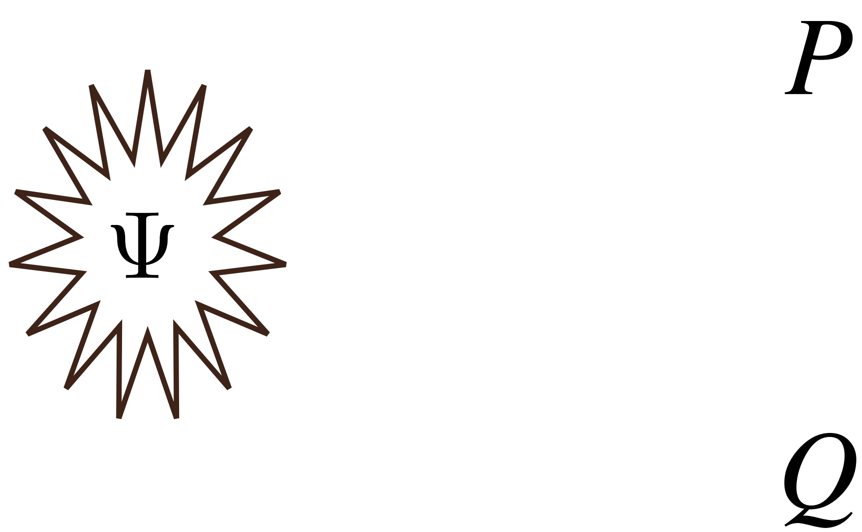
No, since $T \mid \mathbf{0}$ has no action β

Ergo, $P \dot{\sim} Q$ but not $P \mid T \dot{\sim} Q \mid T$

Bisimulation graphically

In the environment Ψ, P and Q behave similarly.

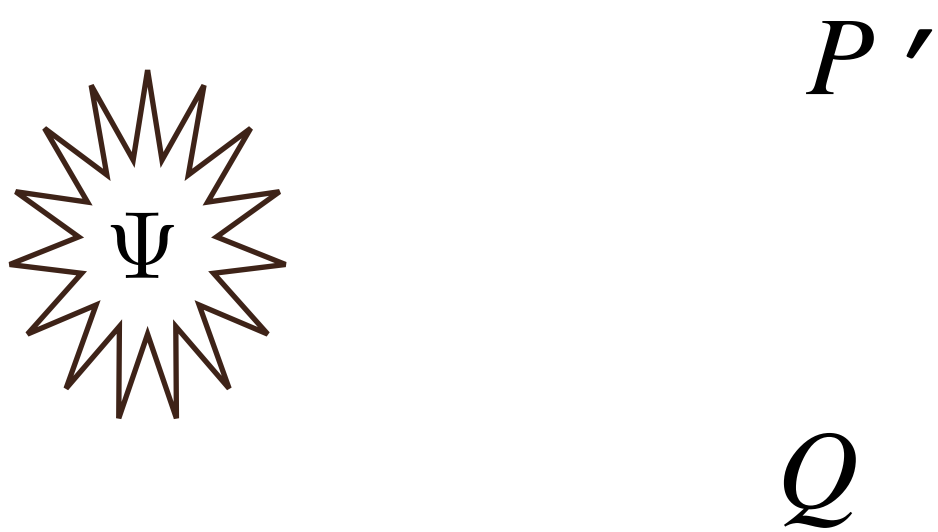
If P' changes to P' then Q can mimic this to Q'



Bisimulation graphically

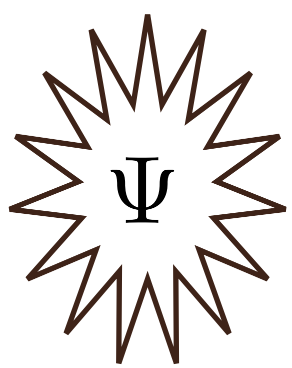
In the environment Ψ, P and Q behave similarly.

If P' changes to P' then Q can mimic this to Q'



Bisimulation graphically

In the environment Ψ, P and Q behave similarly.
If P' changes to P' then Q can mimic this to Q'



P'

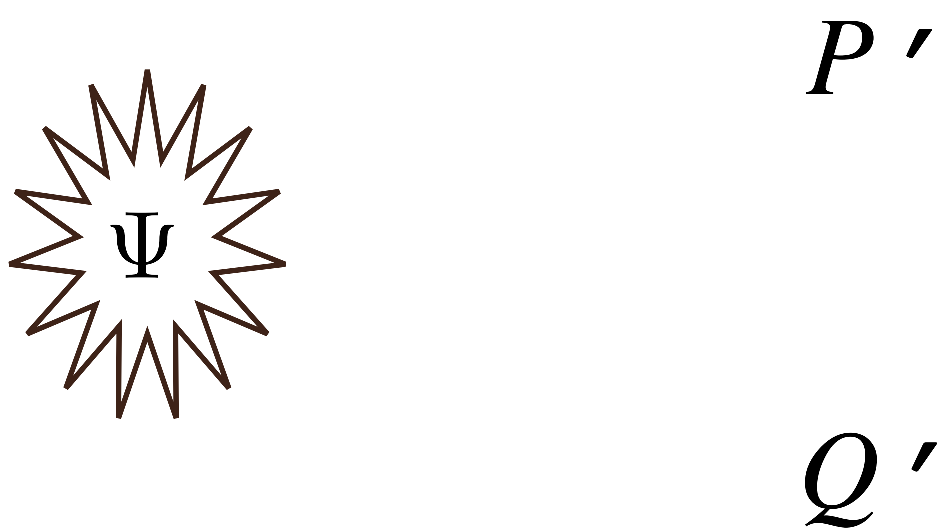
Q'

Bisimulation graphically

In the environment Ψ, P and Q behave similarly.

If P' changes to P' then Q can mimic this to Q'

If the environment changes they should still be bisimilar!

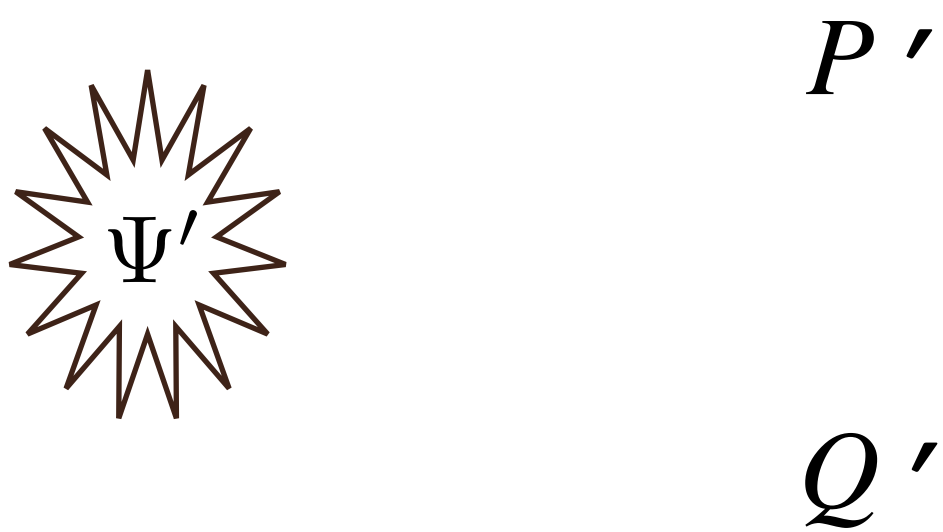


Bisimulation graphically

In the environment Ψ, P and Q behave similarly.

If P' changes to P' then Q can mimic this to Q'

If the environment changes they should still be bisimilar!

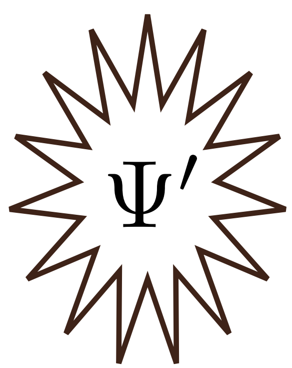


Bisimulation graphically

In the environment Ψ, P and Q behave similarly.

If P' changes to P' then Q can mimic this to Q'

If the environment changes they should still be bisimilar!



P'

Q'

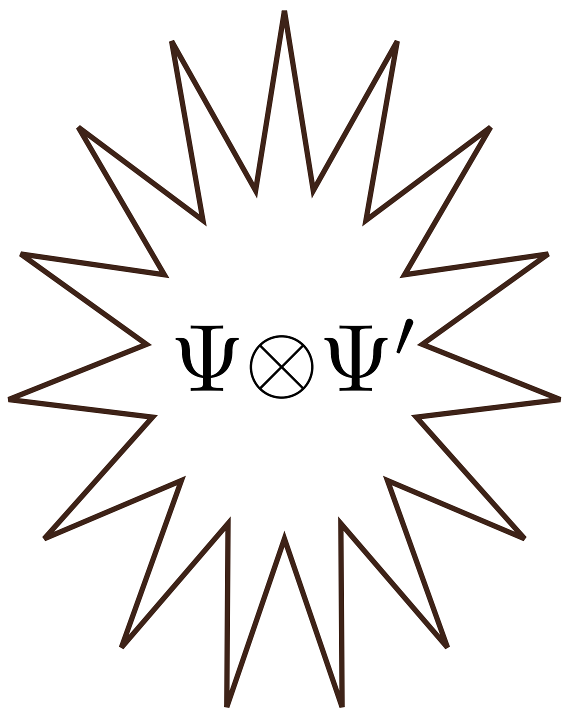
The only way an agent in the environmnent can change the environmental assertion is to add more assertions to it!

Bisimulation graphically

In the environment Ψ, P and Q behave similarly.

If P' changes to P' then Q can mimic this to Q'

If the environment changes they should still be bisimilar!



P'

Q'

The only way an agent in the environmnent can change the environmental assertion is to add more assertions to it!

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$

2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$

   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$

3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha.\mathrm{bn}(\alpha)\#Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$
4. $\forall \Psi'.\ R(\Psi \otimes \Psi', P, Q)$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi.$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$
4. $\forall \Psi'.\ R(\Psi \otimes \Psi', P, Q)$

R is a *bisimulation* if $R(\Psi, P, Q)$ implies

1. $R(\Psi, Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q, \Psi .$
   $\Psi \rhd P \xrightarrow{\alpha} P'$ implies $\Psi \rhd Q \xrightarrow{\alpha} Q'$ and $R(\Psi, P', Q')$
3. $\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$
4. $\forall \Psi' .\ R(\Psi \otimes \Psi', P, Q)$

$\dot{\sim}$ is the largest bisimulation.

$P \sim Q$ If for all sequences of substitutions σ, $P\sigma \mathrel{\dot{\sim}} Q\sigma$

Alternative definition

A binary relation R is a *context bisimulation* if $R(P, Q)$ implies

1. $R(Q, P)$
2. $\forall \alpha . \mathrm{bn}(\alpha) \# Q$
   $\mathbf{1} \rhd P \xrightarrow{\alpha} P'$ implies $\mathbf{1} \rhd Q \xrightarrow{\alpha} Q'$ and $R(P', Q')$
3. $\mathcal{F}(P) \simeq \mathcal{F}(Q)$
4. $\forall \Psi .\ R((\!|\Psi|\!) \mid P, \ \ (\!|\Psi|\!) \mid Q)$

Comparison

$\mathcal{R}(P, Q)$

$\mathcal{R}(\Psi, P, Q)$

Comparison

$\mathcal{R}(P, Q)$

$$P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(P', Q') \end{array}$$

$\mathcal{R}(\Psi, P, Q)$

$$\Psi \rhd P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(\Psi, P', Q') \end{array}$$

Comparison

$\mathcal{R}(P, Q)$

$$P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(P', Q') \end{array}$$

$$\mathcal{F}(P) \simeq \mathcal{F}(Q)$$

$\mathcal{R}(\Psi, P, Q)$

$$\Psi \rhd P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(\Psi, P', Q') \end{array}$$

$$\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$$

Comparison

$$\mathcal{R}(P, Q) \qquad\qquad \mathcal{R}(\Psi, P, Q)$$

$$P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(P', Q') \end{array} \qquad \Psi \rhd P \xrightarrow{\alpha} P' \implies \begin{array}{l} Q \xrightarrow{\alpha} Q' \\ \wedge\, \mathcal{R}(\Psi, P', Q') \end{array}$$

$$\mathcal{F}(P) \simeq \mathcal{F}(Q) \qquad\qquad \Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$$

$$\forall \Psi.\ \mathcal{R}((\!|\Psi|\!) \mid P,\ (\!|\Psi|\!) \mid Q) \qquad\qquad \forall \Psi'.\ \mathcal{R}(\Psi' \otimes \Psi, P, Q)$$

Comparison

$$\mathcal{R}(P, Q) \qquad\qquad \mathcal{R}(\Psi, P, Q)$$

$$P \xrightarrow{\alpha} P' \implies Q \xrightarrow{\alpha} Q' \wedge \mathcal{R}(P', Q') \qquad \Psi \rhd P \xrightarrow{\alpha} P' \implies Q \xrightarrow{\alpha} Q' \wedge \mathcal{R}(\Psi, P', Q')$$

$$\mathcal{F}(P) \simeq \mathcal{F}(Q) \qquad\qquad \Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$$

$$\forall \Psi.\ \mathcal{R}((\!|\Psi|\!) \mid P,\ (\!|\Psi|\!) \mid Q) \qquad\qquad \forall \Psi'.\ \mathcal{R}(\Psi' \otimes \Psi, P, Q)$$

$$\boxed{\mathcal{R}(P, Q) \quad \text{iff} \quad \forall \Psi.\ \mathcal{R}(\Psi, P, Q)}$$

Comparison

$\mathcal{R}(P, Q)$

$\mathcal{R}(\Psi, P, Q)$

$$P \xrightarrow{\alpha} P' \implies Q \xrightarrow{\alpha} Q' \wedge \mathcal{R}(P', Q')$$

$$\Psi \rhd P \xrightarrow{\alpha} P' \implies Q \xrightarrow{\alpha} Q' \wedge \mathcal{R}(\Psi, P', Q')$$

$$\mathcal{F}(P) \simeq \mathcal{F}(Q)$$

$$\Psi \otimes \mathcal{F}(P) \simeq \Psi \otimes \mathcal{F}(Q)$$

$$\forall \Psi.\ \mathcal{R}((\!|\Psi|\!) \mid P,\ (\!|\Psi|\!) \mid Q)$$

$$\forall \Psi'.\ \mathcal{R}(\Psi' \otimes \Psi, P, Q)$$

Manual proof: 1-2 hours, 70 lines
Isabelle proof: 1 day, 450 lines

$$\boxed{\mathcal{R}(P, Q) \quad \text{iff} \quad \forall \Psi.\ \mathcal{R}(\Psi, P, Q)}$$

Results

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Results

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

Results

- Standard Semantics ← Definition of behaviour
- Symbolic Semantics ← More ”computable”
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Results

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

More ”computable”

If P and Q behave the same, then $P|R$ and $Q|R$ behave the same

Results

Efficient proof method

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

More ”computable”

If P and Q behave the same, then $P|R$ and $Q|R$ behave the same

Results

Efficient proof method

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

More "computable"

If P and Q behave the same, then $P|R$ and $Q|R$ behave the same

Intuitive equivalence

Efficient proof method

Results

- Standard Semantics
- Symbolic Semantics
- Compositionality
- Strong and Weak Bisimulation
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

More ”computable”

If P and Q behave the same, then $P|R$ and $Q|R$ behave the same

$$P \mid (Q \mid R) \quad \sim \quad (P \mid Q) \mid R$$

Intuitive equivalence

Efficient proof method

Results

- Standard Semantics
- Sy
- C
- Strong and W ... tion
- Barbed Congruence
- Algebraic Laws

Definition of behaviour

More "computable"

If P and Q behave the same, then $P|R$ and $Q|R$ behave the same

$$P \mid (Q \mid R) \quad \sim \quad (P \mid Q) \mid R$$

Machine checked once and for all

Intuitive equivalence

Correctness: the holy grail

Theory Development in a Theorem Prover

Benefit 1: **Certainty** (no false assertions)
Benefit 2: Good proof **structure** (clarity of arguments)

Theory Development in a Theorem Prover

Benefit 1: **Certainty** (no false assertions)
Benefit 2: Good proof **structure** (clarity of arguments)

Benefit 3: **Flexibility** (easy to change details)
Benefit 4: **Generality** (easy to keep track of assumptions)

Flexibility

- **Theory development is like programming**: It almost never starts from scratch. You continually add, improve, amend, adjust...

Flexibility

- **Theory development is like programming**: It almost never starts from ... nually ... end, adjust...

Please change this one

- **Programming**: Every amendment needs a program recompilation.

- **Theory development**: Every amendment needs a re-check of all proofs. *A huge error source*.

- **Mechanised proofs** means we have a proof repository and can quickly assess ramifications of changes.

Generality

”hmm, some need grease,
some need oil,
I better give both to all...

The lemmas in a theory are like cogwheels, each with a little twisty set of assumptions

...otherwise I would have to remember who needs what.”

Generality

”hmm, some need ~~grease~~, Weakening
some need ~~oil~~, Idempotence
I better give both to all...

The lemmas in a theory are like cogwheels, each with a little twisty set of assumptions

...otherwise I would have to remember who needs what.”

Example from Psi

A binary relation R on agents is an MJbisim if R(P,Q) implies

1. F(P)=F(Q) (static equiv)
2. R(Q,P)
3. Forall Psi. R({Psi}|P, {Psi}|Q)
4. Forall a s.t. bn(a)#Q. P -a-> P' => Q-a->Q' and R(P',Q')

(here transitions without assertion means bottom assertion)

Conjecture 1.
a) Psi |> P -a-> P' implies {Psi}|P -a-> {Psi}|P'.
b) {Psi}|P -a-> T implies exists P'. T = {Psi}|P' and Psi |> P -a-> P'
Proof: For a: by the PAR rule and F({Psi})=Psi. For b: case analysis on deirivation of {Psi}|P-a->T, and here only PAR can be used. Details are left as an exercise for the reader :)

Conjecture 2.
{Psi}|{Psi'} ~ {Psi+Psi'}
Proof: Directly from definitions. Obvious :)

Conjecture 3. If R is an MJbisim up to ~ and R(P,Q) then there is an MJbisim R' such that R'(P,Q)
Proof: By intimidation :)

Lemma 1
If R is an MJbisim then R* =def {(Psi, P,Q): R({Psi}|P, {Psi}|Q)} is a bisimulation up to ~
Proof. We need to check 4 conditions. Assume R*(Psi,P.Q). Then R({Psi}|P, {Psi}|Q).
1. Psi + F(P) = Psi + F(Q). Follows from F({Psi}|P) = F({Psi}|Q).
2. R*(Psi,Q,P). Follows from R({Psi}|Q, {Psi}|P).
3. All Psi' . R*(Psi+Psi', P, Q). Follows from All Psi' . R({Psi'}|{Psi}|P, {Psi'}|{Psi}|Q), and Conjecture 2. Note that here we probably need associativity.
4. Psi |> P-a-> P' implies exists Q' . Psi |> Q-a-> Q' and R(Psi,P',Q'). So assume Psi |> P-a-> P'. Then by Conjecture 1a {Psi}|P -a-> {Psi}|P'. By Condition 4 on MJbisim and R({Psi}|P, {
{Psi}|Q -a-> T with R({Psi}|P',T). Conjecture 1b then gives that there exists a Q' such that T = {Psi}|Q' and {Psi} |> Q -a-> Q'. Also R({Psi}|P',{Psi}|Q') by definition implies R*(Psi,P',Q'), a
QED

Lemma 2.
If R* is a bisimulation then R = def {({Psi}|P, {Psi}|Q): R*(Psi,P,Q)} is an MJbisim up to ~.
Proof. We need to check 4 conditions. Assume R(T,U). By definition there are Psi,P,Q s.t. T={Psi}|P, U={Psi}|Q, R*(Psi,P,Q).
1. F(T)=F(U). Follows from R*(Psi,P,Q) and thus Psi+F(P) = Psi+F(Q).
2. R(U,T). Follows from R*(Psi,Q,P) and definitions.
3. Forall Psi' . R({Psi'}|T, {Psi'}|U). Follows from Forall Psi' . R*(Psi'+Psi,P,Q), Definitions and Conjecture 2.
4. T -a-> T implies exists U' . U -a-> U' and R(T',U'): So assume T -a-> T'. Then by T={Psi}|P and Conjecture 1b we get P' such that Psi |> P -a-> P'. By R*(Psi,P,Q) we get Psi |> Q -a-> Q
R*(Psi,P',Q'). By conjecture 1a we get {Psi}|Q -a-> {Psi}|Q'. So choose U' = {Psi}|Q'. We thus have U -a-> U', and by R*(Psi,P',Q') and definition also R(T',U').
QED

Corollary
P ~ Q iff there exists an MJbisim R such that R(P,Q)
Proof.
=>: Suppose P ~ Q. Then there is a bisimulation R* such that R*(bot,P,Q). Define R as in Lemma 2, using this R*. It follows that R is an MJbisim and R(0|P, 0|Q), and therefore R U {(P
MJ-bisimulation up to ~. By Conjecture 3 there is than an MJbisim as required.
<=: Suppose R is an MJ-bisimulation up to ~ and R(P,Q). Then R(0|P, 0|Q). By Conjecture 3 there is an MJbisim R' such that R'(0|P,0|Q). So by Lemma 1 there is a bisimulation (up to ~
R*(bot,P,Q), which implies P~Q.

Example from Psi

A binary relation R on agents is an MJbisim if R(P,Q) implies

1. F(P)=F(Q) (static equiv)
2. R(Q,P)
3. Forall Psi. R({Psi}|P, {Psi}|Q)
4. Forall a s.t. bn(a)#Q. P -a-> P' => Q-a->Q' and R(P',Q')

(here transitions without assertion means bottom assertion)

Conjecture 1.
a) Psi |> P -a-> P' implies {Psi}|P -a-> {Psi}|P'.
b) {Psi}|P -a-> T implies exists P'. T = {Psi}|P' and Psi |> P -a-> P'
Proof: For a: by the PAR rule and F({Psi})=Psi. For b: case analysis on deirivation of {Psi}|P-a->T, and here only PAR can be used. Details are left as an exercise for the reader :)

Conjecture 2.
{Psi}|{Psi'} ~ {Psi+Psi'}
Proof: Directly from definitions. Obvious :)

Conjecture 3. If R is an MJbisim up to ~ and R(P,Q) then there is an MJbisim R' such that R'(P,Q)
Proof: By intimidation :)

Lemma 1
If R is an MJbisim then R* =def {(Psi, P,Q): R({Psi}|P, {Psi}|Q)} is a bisimulation up to ~
Proof. We need to check 4 conditions. Assume R*(Psi,P.Q). Then R({Psi}|P, {Psi}|Q).
1. Psi + F(P) = Psi + F(Q). Follows from F({Psi}|P) = F({Psi}|Q).
2. R*(Psi,Q,P). Follows from R({Psi}|Q, {Psi}|P).
3. All Psi' . R*(Psi+Psi', P, Q). Follows from All Psi' . R({Psi'}|{Psi}|P, {Psi'}|{Psi}|Q), and Conjecture 2. Note that here we probably need associativity.
4. Psi |> P-a-> P' implies exists Q' . Psi |> Q-a-> Q' and R(Psi,P',Q'). So assume Psi |> P-a-> P'. Then by Conjecture 1a {Psi}|P -a-> {Psi}|P'. By Condition 4 on MJbisim and R({Psi}|P, {
{Psi}|Q -a-> T with R({Psi}|P',T). Conjecture 1b then gives that there exists a Q' such that T = {Psi}|Q' and {Psi} |> Q -a-> Q'. Also R({Psi}|P', {Psi}|Q') by definition implies R*(Psi,P',Q'), a
QED

Lemma 2.
If R* is a bisimulation then R = def {({Psi}|P, {Psi}|Q): R*(Psi,P,Q)} is an MJbisim up to ~.
Proof. We need to check 4 conditions. Assume R(T,U). By definition there are Psi,P,Q s.t. T={Psi}|P, U={Psi}|Q, R*(Psi,P,Q).
1. F(T)=F(U). Follows from R*(Psi,P,Q) and thus Psi+F(P) = Psi+F(Q).
2. R(U,T). Follows from R*(Psi,Q,P) and definitions.
3. Forall Psi' . R({Psi'}|T, {Psi'}|U). Follows from Forall Psi' . R*(Psi'+Psi,P,Q), Definitions and Conjecture 2.
4. T -a> T implies exists U' . U -a-> U' and R(T',U'): So assume T -a-> T'. Then by T={Psi}|P and Conjecture 1b we get P' such that Psi |> P -a-> P'. By R*(Psi,P,Q) we get Psi |> Q -a-> (
R*(Psi,P',Q'). By conjecture 1a we get {Psi}|Q -a-> {Psi}|Q'. So choose U' = {Psi}|Q'. We thus have U -a-> U', and by R*(Psi,P',Q') and definition also R(T',U').
QED

Corollary
P ~ Q iff there exists an MJbisim R such that R(P,Q)
Proof.
=>: Suppose P ~ Q. Then there is a bisimulation R* such that R*(bot,P,Q). Define R as in Lemma 2, using this R*. It follows that R is an MJbisim and R(0|P, 0|Q), and therefore R U {(P
MJ-bisimulation up to ~. By Conjecture 3 there is than an MJbisim as required.
<=: Suppose R is an MJ-bisimulation up to ~ and R(P,Q). Then R(0|P, 0|Q). By Conjecture 3 there is an MJbisim R' such that R'(0|P,0|Q). So by Lemma 1 there is a bisimulation (up to ~
R*(bot,P,Q), which implies P~Q.

Entire manual proof
from email archive
70 lines text
2h work

A binary relation R on agents is an MJbisim if R(P,Q) implies

1. F(P)=F(Q) (static equiv)
2. R(Q,P)
3. Forall Psi. R({Psi}|P, {Psi}|Q)
4. Forall a s.t. bn(a)#Q. P -a-> P' => Q-a->Q' and R(P',Q')
(here transitions without assertion means bottom assertion)

Conjecture 1.
a) Psi |> P -a-> P' implies {Psi}|P -a-> {Psi}|P'.
b) {Psi}|P -a-> T implies exists P'. T = {Psi}|P' and Psi |> P -a-> P'
Proof: For a: by the PAR rule and F({Psi})=Psi. For b: case analysis on deirivation of {Psi}|P

Conjecture 2.
{Psi}|{Psi'} ~ {Psi+Psi'}
Proof: Directly from definitions. Obvious :)

Conjecture 3. If R is an MJbisim up to ~ and R(P,Q) then there is an MJbisim R' such that R
Proof: By intimidation :)

Lemma 1
If R is an MJbisim then R* =def {(Psi, P,Q): R({Psi}|P, {Psi}|Q)} is a bisimulation up to ~
Proof. We need to check 4 conditions. Assume R*(Psi,P.Q). Then R({Psi}|P, {Psi}|Q).
1. Psi + F(P) = Psi + F(Q). Follows from F({Psi}|P) = F({Psi}|Q).

```
lemma bisimContextBisimPar:
  fixes Ψ :: 'b
  and   P  :: "('a, 'b, 'c) psi"
  and   Q  :: "('a, 'b, 'c) psi"

  assumes "Ψ ⊳ P ∼ Q"

  shows "⦃Ψ⦄ ∥ P ∼c ⦃Ψ⦄ ∥ Q"
proof -
  let ?X = "{(⦃Ψ⦄ ∥ P, ⦃Ψ⦄ ∥ Q) | Ψ P Q. Ψ ⊳ P ∼ Q}"
  from assms have "(⦃Ψ⦄ ∥ P, ⦃Ψ⦄ ∥ Q) ∈ ?X" by blast
  thus ?thesis
  proof(coinduct rule: contextBisimWeakCoinduct)
    case(cStatEq P Q)
    thus ?case by(auto dest: bisimE)
  next
    case(cSim P Q)
    have "eqvt ?X" by(force dest: bisimClosed simp add: eqvt_def)
    hence "eqvt({(𝟭, P, Q) | P Q. (P, Q) ∈ ?X})"
      by(auto simp add: eqvt_def permBottom)
    thus ?case using cSim by(blast dest: bisimE intro: contextSimAssertionId)
  next
    case(cExt Ψ PsiP PsiQ)
    from `(PsiP, PsiQ) ∈ ?X` obtain Ψ' P Q where "Ψ' ⊳ P ∼ Q" and A: "PsiP = ⦃Ψ'⦄ ∥ P"
                                                            and B: "PsiQ = ⦃Ψ'⦄ ∥ Q" by auto
    from `Ψ' ⊳ P ∼ Q` have "Ψ' ⊗ Ψ ⊳ P ∼ Q" by(rule bisimE)
    hence "Ψ ⊗ Ψ' ⊳ P ∼ Q" by(metis statEqBisim Commutativity)
    hence "Ψ ⊳ ⦃Ψ'⦄ ∥ P ∼ ⦃Ψ'⦄ ∥ Q" by(rule_tac bisimParPresAuxSym) auto
    with A B show ?case by blast
  next
    case(cSym P Q)
    thus ?case by(blast dest: bisimE)
  qed
qed
```

Corresponding Isabelle proof
475 lines text
8h work

```
lemma bisimContextBisimPar:
  fixes Ψ :: 'b
  and   P  :: "('a, 'b, 'c) psi"
  and   Q  :: "('a, 'b, 'c) psi"

  assumes "Ψ ⊳ P ∼ Q"

  shows "⦃Ψ⦄ ∥ P ∼c ⦃Ψ⦄ ∥ Q"
proof -
  let ?X = "{(⦃Ψ⦄ ∥ P, ⦃Ψ⦄ ∥ Q) | Ψ P Q. Ψ ⊳ P ∼ Q}"
  from assms have "(⦃Ψ⦄ ∥ P, ⦃Ψ⦄ ∥ Q) ∈ ?X" by blast
  thus ?thesis
  proof(coinduct rule: contextBisimWeakCoinduct)
    case(cStatEq P Q)
    thus ?case by(auto dest: bisimE)
  next
    case(cSim P Q)
    have "eqvt ?X" by(force dest: bisimClosed simp add: eqvt_def)
    hence "eqvt({(𝟭, P, Q) | P Q. (P, Q) ∈ ?X})"
      by(auto simp add: eqvt_def permBottom)
    thus ?case using cSim by(blast dest: bisimE intro: contextSimAssertionId)
  next
    case(cExt Ψ PsiP PsiQ)
    from `(PsiP, PsiQ) ∈ ?X` obtain Ψ' P Q where "Ψ' ⊳ P ∼ Q" and A: "PsiP = ⦃Ψ'⦄ ∥ P"
                                                    and B: "PsiQ = ⦃Ψ'⦄ ∥ Q" by auto
    from `Ψ' ⊳ P ∼ Q` have "Ψ' ⊗ Ψ ⊳ P ∼ Q" by(rule bisimE)
    hence "Ψ ⊗ Ψ' ⊳ P ∼ Q" by(metis statEqBisim Commutativity)
    hence "Ψ ⊳ ⦃Ψ'⦄ ∥ P ∼ ⦃Ψ'⦄ ∥ Q" by(rule_tac bisimParPresAuxSym) auto
    with A B show ?case by blast
  next
    case(cSym P Q)
    thus ?case by(blast dest: bisimE)
  qed
qed
```

Different syntax
Same structure

Lemma 2.
If R* is a bisimulation then R = def {({Psi}IP, {Psi}IQ): R*(Psi,P,Q)} is an MJbisim up to ~.
Proof. We need to check 4 conditions. Assume R(T,U). By definition there are Psi,P,Q s.t. T={Psi}IP, U={Psi}IQ, R*(Psi,P,Q).
1. F(T)=F(U). Follows from R*(Psi,P,Q) and thus Psi+F(P) = Psi+F(Q).
2. R(U,T). Follows from R*(Psi,Q,P) and definitions.
3. Forall Psi' . R({Psi'}IT, {Psi'}IU). Follows from Forall Psi' . R*(Psi'+Psi,P,Q), Definitions and Conjecture 2.
4. T -a> T implies exists U' . U -a-> U' and R(T',U'): So assume T -a-> T'. Then by T={Psi}IP and Conjecture 1b we get P' such that Psi I> P -a-> P'. By R*(Psi,P,Q) we get Psi I> Q -a-> Q' and R*(Psi,P',Q'). By conjecture 1a we get {Psi}IQ -a-> {Psi}IQ'. So choose U' = {Psi}IQ'. We thus have U -a-> U', and by R*(Psi,P',Q') and definition also R(T',U').
QED

```
lemma bisimContextBisimPar:
  fixes Ψ :: 'b
  and   P :: "('a, 'b, 'c) psi"
  and   Q :: "('a, 'b, 'c) psi"

  assumes "Ψ ⊳ P ∼ Q"

  shows "⦃Ψ⦄ || P ∼c ⦃Ψ⦄ || Q"
proof -
  let ?X = "{(⦃Ψ⦄ || P, ⦃Ψ⦄ || Q) | Ψ P Q. Ψ ⊳ P ∼ Q}"
  from assms have "(⦃Ψ⦄ || P, ⦃Ψ⦄ || Q) ∈ ?X" by blast
  thus ?thesis
  proof(coinduct rule: contextBisimWeakCoinduct)
    case(cStatEq P Q)
    thus ?case by(auto dest: bisimE)
  next
    case(cSim P Q)
    have "eqvt ?X" by(force dest: bisimClosed simp add: eqvt_def)
    hence "eqvt({((𝟙, P, Q) | P Q. (P, Q) ∈ ?X})"
      by(auto simp add: eqvt_def permBottom)
    thus ?case using cSim by(blast dest: bisimE intro: contextSimAssertionId)
  next
    case(cExt Ψ PsiP PsiQ)
    from `(PsiP, PsiQ) ∈ ?X` obtain Ψ' P Q where "Ψ' ⊳ P ∼ Q" and A: "PsiP = ⦃Ψ'⦄ || P"
                                               and B: "PsiQ = ⦃Ψ'⦄ || Q" by auto
    from `Ψ' ⊳ P ∼ Q` have "Ψ' ⊗ Ψ ⊳ P ∼ Q" by(rule bisimE)
    hence "Ψ ⊗ Ψ' ⊳ P ∼ Q" by(metis statEqBisim Commutativity)
    hence "Ψ ⊳ ⦃Ψ'⦄ || P ∼ ⦃Ψ'⦄ || Q" by(rule_tac bisimParPresAuxSym) auto
    with A B show ?case by blast
  next
    case(cSym P Q)
    thus ?case by(blast dest: bisimE)
  qed
qed
```

The cost?

One measure of effort: **”manhours”**

This particular proof:

Isabelle effort is four times the manual proof

In general

This factor varies wildly

The cost?

One measure of effort: **”manhours”**

Theory development is not exclusively
- not even mainly -
about writing down proofs.

So the factor is not so important.

The cost?

Study of time spent by 4 persons over 25 months on developing the Psi framework

The cost?

Study of time spent by 4 persons over 25 months on developing the Psi framework

1/3 of the effort went into Isabelle formalisation

2/3 of the total effort has been fully formalised