Enhancements of the bisimulation proof method

Davide Sangiorgi

Focus Lab.,
INRIA (France) and University of Bologna (Italy)

Email: Davide.Sangiorgi@cs.unibo.it
http://www.cs.unibo.it/~sangio/

BASICS 2009, Shanghai

CONTENTS

Equality on processes, coinductively

Bisimulation:

A relation $\mathcal{R}$ s.t.
$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \mathcal{R} & Q' \end{array}$$

Bisimilarity ($\sim$) :

$$\bigcup \{\mathcal{R} : \mathcal{R} \text{ is a bisimulation } \}$$

Hence:

$$\frac{x \,\mathcal{R}\, y \qquad \mathcal{R} \text{ is a bisimulation}}{x \sim y}$$

(bisimulation proof method)

This talk

Enhancements of the bisimulation proof method

- Motivations
- Results and Examples
- Open problems

CONTENTS

Examples, in CCS-like calculus

Pieces of syntax and transitions

inp: $a.P \xrightarrow{a} P$

rep: $$\frac{P \mid !P \xrightarrow{a} P'}{!P \xrightarrow{a} P'}$$

sumL: $$\frac{P \xrightarrow{a} P'}{P + Q \xrightarrow{a} P'}$$

sumR: $$\frac{Q \xrightarrow{a} Q'}{P + Q \xrightarrow{a} Q'}$$

parL: $$\frac{P \xrightarrow{a} P'}{P \mid Q \xrightarrow{a} P' \mid Q}$$

parR: $$\frac{Q \xrightarrow{a} Q'}{P \mid Q \xrightarrow{a} P \mid Q'}$$

Intuitively: $!P = P \mid P \mid \ldots \mid P$

(indeed $P \mid !P \sim !P$, as the transitions are the same)

Process abbreviations: $a \triangleq a.0$

$P^n \triangleq P \mid \ldots \mid P$ (n times)

Enhancements of the bisimulation method: an example

A property of replication

$$!(a + b) \sim\ !a \mid !b$$

Proof: Let's find a bisimulation...

Is this a bisimulation?

$$\mathcal{R} \triangleq \{ \, (\boldsymbol{!}(a + b) \, , \, \boldsymbol{!}a \mid \boldsymbol{!}b) \, \}$$

Is this a bisimulation?

$$\mathcal{R} \triangleq \{ \, (!(a+b) \, , \, !a \mid !b) \, \}$$

No!

$$!(a+b) \xrightarrow{a} (a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)$$

$$\mathcal{R} \qquad\qquad \cancel{\mathcal{R}}$$

$$!a \mid !b \xrightarrow{a} a^n \mid 0 \mid a^m \mid !a \mid !b$$

NB: $\forall n, m, \quad !(a+b) \xrightarrow{a} (a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)$

$$!a \mid !b \xrightarrow{a} a^n \mid 0 \mid a^m \mid !a \mid !b$$

Try again...

Is this a bisimulation?

$$\mathcal{R} \triangleq \bigcup_{n,m}$$

$$\{ \quad ((a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)\ ,\ a^n \mid 0 \mid a^m \mid !a \mid !b)\ ,$$

$$((a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)\ ,\ !a \mid b^n \mid 0 \mid b^m \mid !b) \quad \}$$

Is this a bisimulation?

$$\mathcal{R} \triangleq \bigcup_{n,m}$$

$$\{ \quad ((a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)\ ,\ a^n \mid 0 \mid a^m \mid !a \mid !b)\ ,$$

$$((a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b)\ ,\ !a \mid b^n \mid 0 \mid b^m \mid !b) \quad \}$$

No!

$$(a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b) \xrightarrow{b} (a+b)^{n-1} \mid 0^2 \mid (a+b)^m \mid !(a+b)$$

$$\mathcal{R} \qquad \qquad \not\mathcal{R}$$

$$a^n \mid 0 \mid a^m \mid !a \mid !b \xrightarrow{b} a^n \mid 0 \mid a^m \mid !a \mid b \mid !b$$

Try again...

It is possible to write the full bisimulation, but one has to be careful

We started with the **singleton** relation

$$\{(!(a+b)\ ,\ !a \mid !b)\}$$

The added pairs: **redundant**? (derivable, laws of $\sim$)

Can we work with relations smaller than bisimulations?

Advantages:

- fewer and simpler bisimulation diagrams
- easier to find the relation to work with

Redundant pairs

What we would like to have:

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \mathcal{R} \cup \{\textbf{redundant pairs}\} & Q' \end{array} \quad \text{implies} \quad \mathcal{R} \subseteq \sim$$

$\mathcal{R}$: less work, simpler to find

Notation

$$\mathcal{R} \rightarrowtail \mathcal{S} \triangleq \begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \mathcal{S} & Q' \end{array}$$

Up-to techniques: example

– **Rules for transitivity of $\sim$ (up-to $\sim$)**

$$\mathcal{R} \rightarrowtail \sim \mathcal{R} \sim \text{ implies } \mathcal{R} \subseteq \sim$$

diagram :

$$\begin{array}{ccccccc}
P & & & \mathcal{R} & & & Q \\
\alpha \downarrow & & & & & & \downarrow \alpha \\
P' & \sim & P'' & \mathcal{R} & Q'' & \sim & Q'
\end{array}$$

Now we can establish $!(a+b) \sim !a \mid !b$ using the singleton relation

$$\mathcal{R} \triangleq \{!(a+b), !a \mid !b\}$$

and proving that it is a bisimulation up-to $\sim$

$$\begin{array}{ccc}
!(a+b) & \xrightarrow{a} & (a+b)^n \mid 0 \mid (a+b)^m \mid !(a+b) \\
 & & \sim \\
 & & !(a+b) \\
\mathcal{R} & & \mathcal{R} \\
 & & !a \mid !b \\
 & & \sim \\
!a \mid !b & \xrightarrow{a} & a^n \mid 0 \mid a^m \mid !a \mid !b
\end{array}$$

(laws $P \mid !P \sim !P$ and $0 \mid P \sim P$, congruence of $\sim$)

A more interesting example

$$!(a.\,P + b.\,Q) \sim\ !a.\,P \mid\ !b.\,Q$$

Proof: Let's find a bisimulation...

Is this a bisimulation up-to $\sim$?

$$\mathcal{R} \triangleq \{ \, (!(a.\,P + b.\,Q) \, , \; !a.\,P \mid !b.\,Q) \, \}$$

Is this a bisimulation up-to $\sim$?

$$\mathcal{R} \triangleq \{ \, (!(a.\, P + b.\, Q) \, , \; !a.\, P \mid !b.\, Q) \, \}$$

No!

$$
\begin{array}{ccc}
!(a.\, P + b.\, Q) & \xrightarrow{a} \sim & P \mid !(a.\, P + b.\, Q) \\
\mathcal{R} & & \cancel{\mathcal{R}} \\
!a.\, P \mid !b.\, Q & \xrightarrow{a} \sim & P \mid !a.\, P \mid !b.\, Q
\end{array}
$$

- Note also, if $P \xrightarrow{c} P'$:

$$!(a.\,P + b.\,Q) \quad \xrightarrow{a}\xrightarrow{a}\xrightarrow{c} \quad P' \mid P \mid !(a.\,P + b.\,Q)$$

$$!a.\,P \mid !b.\,Q \quad \xrightarrow{a}\xrightarrow{a}\xrightarrow{c} \quad P' \mid P \mid !a.\,P \mid !b.\,Q$$

so a bisimulation up-to $\sim$ should include also such pairs of derivates

- Again, these added pairs may be considered redundant (for instance, $!(a.\,P + b.\,Q) \sim !a.\,P \mid !b.\,Q$ implies $P' \mid P \mid !(a.\,P + b.\,Q) \sim P' \mid P \mid !a.\,P \mid !b.\,Q$)

- We can avoid these additional pairs using a different form of up-to

Up-to techniques: example

- Rules for transitivity of $\sim$ (up-to $\sim$)
- **rules for substitutivity of $\sim$ (up-to context)**

$$\mathcal{C}(\mathcal{R}) \triangleq \{(C[P], C[Q]) \ : \ P \ \mathcal{R} \ Q\}$$

$\mathcal{R} \rightarrowtail \mathcal{C}(\mathcal{R})$ implies $\mathcal{R} \subseteq \sim$

diagram :

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha \downarrow & & \downarrow \alpha \\ \cancel{C}\,[P'] & \mathcal{R} & \cancel{C}\,[Q'] \end{array}$$

Example of composition of techniques

We can put together up-to $\sim$ and up-to context

$$\mathcal{R} \rightarrowtail \sim C[\mathcal{R}] \sim \text{ implies } \mathcal{R} \subseteq \sim$$

diagram :

$$\begin{tikzcd}
P \arrow[d, "\alpha"'] & & \mathcal{R} & & Q \arrow[d, "\alpha"] \\
P' & \sim & \cancel{C}[P''] \;\mathcal{R}\; \cancel{C}[Q''] & \sim & Q'
\end{tikzcd}$$

Back to our proof:

$$\mathcal{R} \triangleq \{!(a.\,P + b.\,Q),\, !a.\,P \mid !b.\,Q\}$$

is a bisimulation up-to $\sim$ and up-to context

$$!(a.\,P + b.\,Q) \xrightarrow{a} (a.\,P + b.\,Q)^n \mid P \mid (a.\,P + b.\,Q)^m \mid !(a.\,P + b.\,Q)$$

$$\mathcal{R}$$

$$!a.\,P \mid !b.\,Q \xrightarrow{a} (a.\,P)^n \mid P \mid (a.\,P)^m \mid !a.\,P \mid !b.\,Q$$

Back to our proof:

$$\mathcal{R} \triangleq \{!(a.\,P + b.\,Q), !a.\,P \mid !b.\,Q\}$$

is a bisimulation up-to $\sim$ and up-to context

$$!(a.\,P + b.\,Q) \xrightarrow{a} (a.\,P + b.\,Q)^n \mid P \mid (a.\,P + b.\,Q)^m \mid !(a.\,P + b.\,Q)$$

$$\sim$$

$$P \mid !(a.\,P + b.\,Q)$$

$$\mathcal{R}$$

$$P \mid !a.\,P \mid !b.\,Q$$

$$\sim$$

$$!a.\,P \mid !b.\,Q \xrightarrow{a} (a.\,P)^n \mid P \mid (a.\,P)^m \mid !a.\,P \mid !b.\,Q$$

Back to our proof:

$$\mathcal{R} \triangleq \{!(a.\,P + b.\,Q), !a.\,P \mid !b.\,Q\}$$

is a bisimulation up-to $\sim$ and up-to context

$$!(a.\,P + b.\,Q) \xrightarrow{a} (a.\,P + b.\,Q)^n \mid P \mid (a.\,P + b.\,Q)^m \mid !(a.\,P + b.\,Q)$$

$$\sim$$

$$\cancel{P} \mid !(a.\,P + b.\,Q)$$

$$\mathcal{R}$$

$$\cancel{P} \mid !a.\,P \mid !b.\,Q$$

$$\sim$$

$$!a.\,P \mid !b.\,Q \xrightarrow{a} (a.\,P)^n \mid P \mid (a.\,P)^m \mid !a.\,P \mid !b.\,Q$$

Back to our proof:

$$\mathcal{R} \triangleq \{!(a.\,P + b.\,Q), !a.\,P \mid !b.\,Q\}$$

is a bisimulation up-to $\sim$ and up-to context

$$!(a.\,P + b.\,Q) \xrightarrow{a} (a.\,P + b.\,Q)^n \mid P \mid (a.\,P + b.\,Q)^m \mid !(a.\,P + b.\,Q)$$

$$\sim$$

$$\cancel{P} \mid !(a.\,P + b.\,Q)$$

$$\mathcal{R} \qquad \mathcal{R}$$

$$\cancel{P} \mid !a.\,P \mid !b.\,Q$$

$$\sim$$

$$!a.\,P \mid !b.\,Q \xrightarrow{a} (a.\,P)^n \mid P \mid (a.\,P)^m \mid !a.\,P \mid !b.\,Q$$

More up-to techniques: example

- Rules for transitivity of $\sim$ (up-to $\sim$)
- rules for substitutivity of $\sim$ (up-to context)
- **rules for invariance of $\sim$ under injective substitutions (up-to injective substitutions)**

$$\textbf{Inj}(\mathcal{R}) \triangleq \{(P\sigma, Q\sigma) \; : \; P \; \mathcal{R} \; Q \, , \;\; \sigma \text{ injective on names}\}$$

$$\boxed{\mathcal{R} \rightarrowtail \textbf{Inj}(\mathcal{R}) \text{ implies } \mathcal{R} \subseteq \sim}$$

diagram :

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha \downarrow & & \downarrow \alpha \\ P'\cancel{\sigma} & \mathcal{R} & Q'\cancel{\sigma} \end{array}$$

σ: an injective function

implies $\quad \mathcal{R} \subseteq \sim$

More composition of techniques

$\mathcal{R} \rightarrowtail \sim C[\textbf{Inj}(\mathcal{R})] \sim$ implies $\mathcal{R} \subseteq \sim$

diagram :

$$
\begin{tikzcd}
P \arrow[d, "\alpha"'] & & & \mathcal{R} & & & Q \arrow[d, "\alpha"] \\
P' & \sim & \cancel{C}[P'' \cancel{\sigma}] & \mathcal{R} & \cancel{C}[Q'' \cancel{\sigma}] & \sim & Q'
\end{tikzcd}
$$

More sophistication $\Rightarrow$

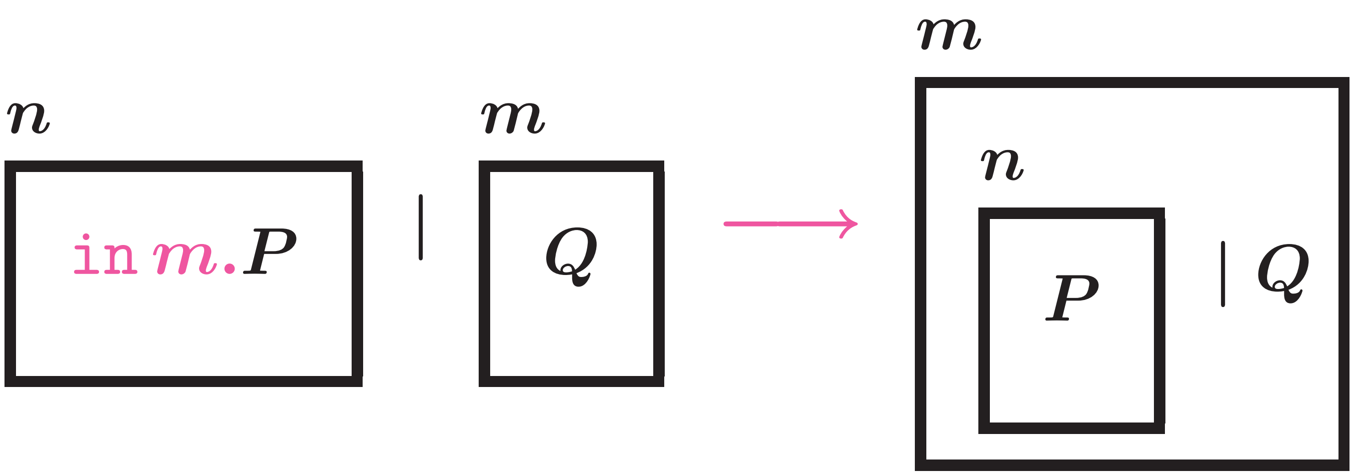
- more powerful technique
- harder soundness proof for the technique

More examples, in a higher-order calculus (the Ambient calculus)

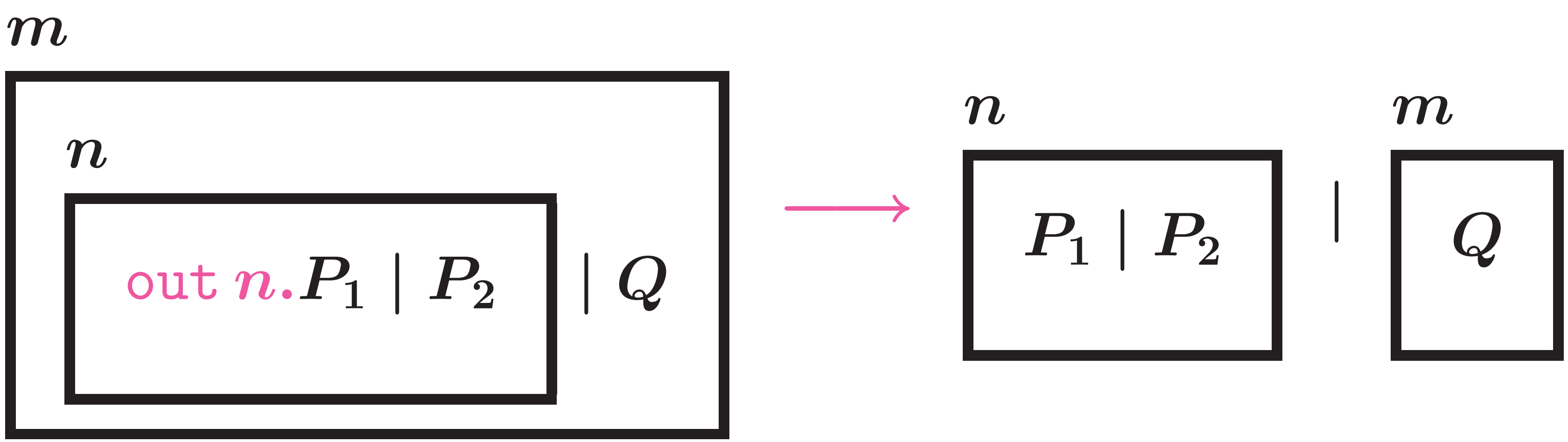
Ambients: syntax

			Processes
P	$::=$	$n\langle P\rangle$	ambient
	$\mid$	$\mathtt{in}\, n.\, P$	in action
	$\mid$	$\mathtt{out}\, n.\, P$	out action
	$\mid$	$\mathtt{open}\, n.\, P$	open action
	$\mid$	$P \mid P$	parallel
	$\mid$	$\nu n\; P$	restriction
	$\mid$	$\ldots$	

The `in` movement



The `out` movement



Example property

The perfect-firewall equation in Ambients

P: a process with n not free in it

$$\nu n \;\; n\langle P\rangle \sim \mathbf{0}$$

Proof: **Let's find a bisimulation...**

Is this a bisimulation?

$$\mathcal{R} \triangleq \{ (\nu n\ n\langle P\rangle\ ,\ 0) \}$$

Is this a bisimulation?

$$\mathcal{R} \triangleq \{\ (\nu n\ n\langle P\rangle\ ,\ 0)\ \}$$

No!

Suppose $n\langle P\rangle \xrightarrow{\mathtt{enter\_}k\langle Q\rangle} n\langle P\rangle$

(the loop: simplifies the example, not necessary)

$$\begin{array}{ccc}
\nu n\ n\langle P\rangle & \mathcal{R} & 0 \\
\mathtt{enter_}k\langle Q\rangle \downarrow & & \downarrow \mathtt{enter_}k\langle Q\rangle \\
k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & \cancel{\mathcal{R}} & k\langle Q\rangle \mid 0
\end{array}$$

Try again...

Is this a bisimulation?

$$\mathcal{R} \triangleq \{ (\nu n\ n\langle P\rangle\ ,\ 0) \}$$

$$\bigcup_{k,Q} \{ (k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle\ ,\ k\langle Q\rangle \mid 0) \}$$

Is this a bisimulation?

$$\mathcal{R} \triangleq \{ (\nu n\ n\langle P\rangle\ ,\ 0) \} \cup_{k,Q} \{ (k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle\ ,\ k\langle Q\rangle \mid 0) \}$$

No!

Suppose $Q = h\langle \mathtt{out}\, k.\, R\rangle \mid Q'$

$$k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle \quad \mathcal{R} \quad k\langle Q\rangle \mid 0$$

$$\downarrow \qquad\qquad \downarrow$$

$$k\langle Q' \mid \nu n\ n\langle P\rangle\ \rangle \mid h\langle R\rangle \quad \cancel{\mathcal{R}} \quad k\langle Q'\rangle \mid h\langle R\rangle \mid 0$$

Try again...

Is this a bisimulation?

$$\mathcal{R} \triangleq \{\ (\nu n\ n\langle P\rangle\ ,\ 0)\ \}$$
$$\bigcup_{k,Q}\ \{\ (k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle\ ,\ k\langle Q\rangle \mid 0)\ \}$$

Also:

Suppose $Q = \mathtt{in}\, h.\, Q'$

$$\begin{array}{ccc} k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & \mathcal{R} & k\langle Q\rangle \mid 0 \\ \mathtt{enter\_}h\langle R\rangle \downarrow & & \downarrow \mathtt{enter\_}h\langle R\rangle \\ h\langle R \mid k\langle Q' \mid \nu n\ n\langle P\rangle\ \rangle\ \rangle & \cancel{\mathcal{R}} & h\langle R \mid k\langle Q'\ \rangle\ \rangle \mid 0 \end{array}$$

Try again...

The bisimulation:

$$\mathcal{R} \triangleq \bigcup_{C \text{ is a static contexts}} \{(S,T) : \begin{array}{lll} S & \sim & C[\nu n\ n\langle P\rangle\ ] \\ T & \sim & C[0] \end{array} \}$$

$$C ::= k\langle C\rangle \ \big|\ P \mid C \ \big|\ \nu a\, C \ \big|\ [\,]$$

We started with the **singleton** relation

$$\{(\nu n\ n\langle P\rangle\ ,\ 0)\}$$

The added pairs: **redundant**? (derivable, laws of $\sim$)

Proof of the firewall, composition of up-to techniques

We can prove $\nu n\ n\langle P\rangle \ \sim\ \mathbf{0}$ using the singleton relation

$$\begin{array}{ccc} \nu n\ n\langle P\rangle & \mathcal{R} & \mathbf{0} \\ \texttt{enter\_}k\langle Q\rangle \downarrow & & \downarrow \texttt{enter\_}k\langle Q\rangle \\ k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & & k\langle Q\rangle \mid \mathbf{0} \end{array}$$

Proof of the firewall, composition of up-to techniques

We can prove $\nu n\ n\langle P\rangle \sim \mathbf{0}$ using the singleton relation

$$
\begin{array}{ccc}
\nu n\ n\langle P\rangle & \mathcal{R} & \mathbf{0} \\
\mathtt{enter\_}k\langle Q\rangle \downarrow & & \downarrow \mathtt{enter\_}k\langle Q\rangle \\
k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & & k\langle Q\rangle \mid \mathbf{0} \\
\sim & & \sim \\
k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & & k\langle Q \mid \mathbf{0}\rangle
\end{array}
$$

Proof of the firewall, composition of up-to techniques

We can prove $\nu n\ n\langle P\rangle\ \sim\ \mathbf{0}$ using the singleton relation

$$
\begin{array}{ccc}
\nu n\ n\langle P\rangle & \mathcal{R} & \mathbf{0} \\
\Big\downarrow{\scriptstyle \mathtt{enter\_}k\langle Q\rangle} & & \Big\downarrow{\scriptstyle \mathtt{enter\_}k\langle Q\rangle} \\
k\langle Q \mid \nu n\ n\langle P\rangle\ \rangle & & k\langle Q\rangle \mid \mathbf{0} \\
\sim & & \sim \\
k\cancel{\langle Q}\ |\nu n\ n\langle P\rangle\ \rangle & \mathcal{R} & k\cancel{\langle Q} \mid \mathbf{0}\rangle
\end{array}
$$

[Zappa-Nardelli, Merro, JACM]

"up-to $\sim$" **and** "up-to context"

(full proof also needs up-to injective substitutions)

Conclusions, part I

- **Enhancements of the bisimulation proof methods: extremely useful**
  - **essential** in π-calculus-like languages, higher-order languages
- **Various forms of enhancement ("up-to techniques") exist**
  - composition of techniques
- **Proofs of soundness of these techniques may be non-trivial**
  - separate ad hoc proofs for each technique

CONTENTS

Redundant pairs: first attempt

$\mathcal{S} \triangleq$ a set of inference rules valid for $\sim$

(P, Q) **redundant** in $\{(P, Q)\} \cup \mathcal{R}$ if

$$\mathcal{S} \quad \frac{\mathcal{R} \subseteq \sim}{P \sim Q}$$

Sound ? i.e.:

$$\begin{tikzcd} P \arrow[d, "\alpha"'] & \mathcal{R} & Q \arrow[d, "\alpha"] \\ P' & \mathcal{R} \cup \{\textbf{redundant pairs}\} & Q' \end{tikzcd}$$

implies $\mathcal{R} \subseteq \sim$

?

False!

Counterexample (in CCS)

$$\mathcal{S} \triangleq \frac{a.\,P \sim a.\,Q}{P \sim Q}$$

$$\mathcal{R} \quad \triangleq \quad \{(a.\,b, a.\,c)\}$$

$\{(b,c)\}$ redundant in $\mathcal{R} \cup \{(b,c)\}$

$$\mathcal{S} \quad \frac{\mathcal{R} \subseteq \sim}{b \sim c}$$

$$\begin{array}{ccc} a.\,b & \mathcal{R} & a.\,c \\ a \downarrow & & \downarrow a \\ b & \mathcal{R} \cup \{(b,c)\} & c \end{array}$$

but $a.\,b \not\sim a.\,c$

Another example

Recall the “bisimulation up-to context and up-to $\sim$” technique:

diagram :

$$\begin{tikzcd}
P \arrow[d, "\alpha"'] & & & \mathcal{R} & & & Q \arrow[d, "\alpha"] \\
P' & \sim & \cancel{\mathcal{C}}[P''] & \mathcal{R} & \cancel{\mathcal{C}}[Q''] & \sim & Q'
\end{tikzcd}$$

It seems valid because $\sim$ is transitive and is a congruence,
hence usable in all languages where $\sim$ has these 2 properties

False!

$$P \;:=\; f(P) \;\big|\; a.\,P \;\big|\; 0$$

$$\frac{P \xrightarrow{a} P' \qquad P' \xrightarrow{a} P''}{f(P) \xrightarrow{a} P''}$$

Bisimulation is a congruence, yet:

$$\begin{tikzcd}
a.\,0 \arrow[d, "a"'] & & \mathcal{R} & & & a.\,a.\,0 \arrow[d, "a"] \\
0 & \sim\; f(a.\,0) & & & f(a.\,a.\,0) \;\sim & a.\,0
\end{tikzcd}$$

False!

$$P \;::=\; f(P) \;\Big|\; a.\,P \;\Big|\; 0$$

$$\frac{P \xrightarrow{a} P' \qquad P' \xrightarrow{a} P''}{f(P) \xrightarrow{a} P''}$$

Bisimulation is a congruence, yet:

$$\begin{tikzcd}
a.0 \arrow[d, "a"'] & & \mathcal{R} & & a.a.0 \arrow[d, "a"] \\
0 & \sim \cancel{f}(a.\,0) & \mathcal{R} & \cancel{f}(a.\,a.\,0) \sim & a.0
\end{tikzcd}$$

Weak bisimilarity ($\approx$)

$$\Longrightarrow \quad \triangleq \quad \longrightarrow \cdots \longrightarrow$$

$$\overset{\alpha}{\Longrightarrow} \quad \triangleq \quad \Longrightarrow \xrightarrow{\alpha} \Longrightarrow$$

Weak bisimulation

Too heavy:
$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha \Downarrow & & \Downarrow \alpha \\ P' & \mathcal{R} & Q' \end{array}$$

Better:

(read: $\Rightarrow$)

$$\boxed{\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha \downarrow & & \Downarrow \alpha \\ P' & \mathcal{R} & Q' \end{array}}$$

Example: up-to bisimilarity that fails

$\approx$ **is transitive, yet:**

$$
\begin{array}{ccc}
\tau.\,a.\,0 & \mathcal{R} & 0 \\
\downarrow & & \Downarrow \\
a.\,0 & & 0 \\
\approx & & \approx \\
\tau.\,a.\,0 & \mathcal{R} & 0
\end{array}
$$

Conclusions, part II

– **When is a pair redundant?**

– **Needed: a general theory of enhancements of the bisimulation proof method**

  * powerful techniques
  * combination of techniques
  * easy to derive their soundness

– **An attempt: sound functions, respectful functions**

[Sangiorgi]

– NB: all results that follow proved in Coq

CONTENTS

Sound functions

$\mathcal{F} : \wp(\mathcal{P} \times \mathcal{P}) \mapsto \wp(\mathcal{P} \times \mathcal{P})$:

sound if $\mathcal{R} \rightarrowtail \mathcal{F}(\mathcal{R})$ implies $\mathcal{R} \subseteq \sim$

$$\begin{tikzcd} P \arrow[d, "\alpha"'] & \mathcal{R} & Q \arrow[d, "\alpha"] \\ P' & \mathcal{F}(\mathcal{R}) & Q' \end{tikzcd}$$

Each sound function: a valid enhancement

- **Are there interesting sound functions?**
- **Properties:**
  - ∗ membership easy to check?
  - ∗ nice compositionality properties?

Sound functions

$\mathcal{F} : \wp(\mathcal{P} \times \mathcal{P}) \mapsto \wp(\mathcal{P} \times \mathcal{P})$:

sound if $\mathcal{R} \rightarrowtail \mathcal{F}(\mathcal{R})$ implies $\mathcal{R} \subseteq \sim$

$$\begin{tikzcd} P \arrow[d, "\alpha"'] & \mathcal{R} & Q \arrow[d, "\alpha"] \\ P' & \mathcal{F}(\mathcal{R}) & Q' \end{tikzcd}$$

Each sound function: a valid enhancement

- **Are there interesting sound functions? YES**
- **Properties:**
  - ∗ membership easy to check? **NO**
  - ∗ nice compositionality properties? **NO**

Towards an algebra of up-to techniques

$\mathcal{F}$: Relations $\mapsto \wp(\mathcal{P} \times \mathcal{P})$

respectful if $\dfrac{\mathcal{R} \subseteq \mathcal{S} \qquad \mathcal{R} \rightarrowtail \mathcal{S}}{\mathcal{F}(\mathcal{R}) \subseteq \mathcal{F}(\mathcal{S}) \qquad \mathcal{F}(\mathcal{R}) \rightarrowtail \mathcal{F}(\mathcal{S})}$

Examples:

identity	$\mathcal{I}$	$[\mathcal{I}(\mathcal{R}) = \mathcal{R}]$
constant-to $\sim$	$\mathcal{U}$	$[\mathcal{U}(\mathcal{R}) = \sim]$
closure under monadic contexts	$\mathcal{C}$	
closure under inj. substitutions	$\mathbf{Inj}$	

Proofs of respectfulness: easy

Non-example: **constant-to** $\mathcal{P} \times \mathcal{P}$

Compositionality properties

A respectful second-order function:
preserves the respectfulness of its arguments

Examples:

composition	$\circ$	$[\ \ (\mathcal{G} \circ \mathcal{F})\langle\mathcal{R}\rangle = \mathcal{G}\langle\mathcal{F}\langle\mathcal{R}\rangle\rangle\ \ ]$
union	$\bigcup_{i \in I}$	$[\ \ (\bigcup_{i \in I} \mathcal{F}_i)\langle\mathcal{R}\rangle \triangleq \bigcup_{i \in I}(\mathcal{F}_i\langle\mathcal{R}\rangle)\ \ ]$
chaining	$\frown$	$[\ \ (\mathcal{G} \frown \mathcal{F})\langle\mathcal{R}\rangle = \mathcal{G}(\mathcal{R})\ \mathcal{F}(\mathcal{R})\ \ ]$

Proofs of respectfulness: easy

The previous up-to techniques before can be derived:

$$\mathcal{U}^\frown\mathcal{I}^\frown\mathcal{U} \quad = \quad \text{up-to } \sim \qquad \begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \sim\mathcal{R}\sim & Q' \end{array}$$

$$\bigcup_{n>0} \underbrace{\mathcal{I}^\frown\cdots^\frown\mathcal{I}}_{n} \quad = \quad \text{up-to transitive closure} \qquad \begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \mathcal{R}^+ & Q' \end{array}$$

Similarly we derive: $\mathcal{C}^*$ (up-to **polyadic** contexts)

$$\sim \mathcal{C}^*(\mathbf{Inj}(\mathcal{R})) \sim$$

Conclusions, part III

- **An attempt of an algebra of enhancements**
  - Minimal basic ingredients

    (identity function, constant functions, ...)
  - 2nd order functions to derive more powerful techniques
- **Sufficient to derive many techniques of practical interest**

  (for strong bisimulation)
- **However**, in this theory:
  - ad hoc definitions?
  - all proofs very easy

Problem 1: Robust definition of enhancement

- **Better definition of respectfulness ?**
- **Abstract formulations of a more powerful bisimulation principle ?**
- **Generalisation to coinduction ?**
  - Partial results on coalgebras **[Lenisa, Honsell]**

Problem 2: soundness of up-to context

- **What conditions on contexts for the up-to context to be sound?**
  * Bisimulation as a congruence? i.e.:

$$
\begin{array}{ccc}
P & \mathcal{R} & Q \\
\alpha \downarrow & & \downarrow \alpha \\
\cancel{C}[P'] & \mathcal{R} & \cancel{C}[Q']
\end{array}
$$

sound iff C preserves $\sim$

?

- **And for respectfulness?**
  * Bisimulation as a congruence? **No!**
  * Partial answer: some behavioural conditions on contexts **[Sangiorgi]**

Problem 3: up-to context in higher-order languages

Example: λ-calculus (call-by-value/name,typed/untyped,...)

$$M \xRightarrow{\lambda R} N \quad \triangleq \quad M \Longrightarrow \lambda x.\, M' \quad \& \quad N = M'\{ {}^{R}/x \}$$

Applicative bisimulation ($\simeq$) :

$$\begin{array}{ccc} M & \mathcal{R} & N \\ \lambda R \Downarrow & & \Downarrow \lambda R \\ M' & \mathcal{R} & N' \end{array}$$

Theorem: $\simeq$ is a congruence

Applicative bisimulation up-to polyadic contexts

$$\begin{array}{ccc} M & \mathcal{R} & N \\ \lambda R \Downarrow & & \Downarrow \lambda R \\ \cancel{C}[M_1, \ldots, M_n] & \mathcal{R} & \cancel{C}[N_1, \ldots, N_n] \end{array}$$

implies $\mathcal{R} \subseteq \simeq$?

- Related to the problem of **compositionality** of bisimulation?
- Soundness of **limited forms of up-to context** :
  * **[Pitts 96, Lassen 98]**: typed and untyped λ-calculus, for various reduction strategies
  * **[Koutavas, Wand 06]**: a λ-calculus with references

Example of use:

Park-induction property for various fixed-point combinators (Curry, Turing, call-by-value, rec)

$$\frac{\lambda x.\, e\{ {}^{v}/f \} \lesssim v}{\mathtt{rec} f = \lambda x.\, e \lesssim v}$$

CONTENTS

Example: up-to bisimilarity that fails

$$\begin{array}{ccc} \tau.\,a.\,0 & \mathcal{R} & 0 \\ \downarrow & & \Downarrow \\ a.\,0 & & 0 \\ \approx & & \approx \\ \tau.\,a.\,0 & \mathcal{R} & 0 \end{array}$$

Example: up-to bisimilarity that fails

$$
\begin{array}{ccc}
\tau.\,a.\,0 & \mathcal{R} & 0 \\
\downarrow & & \Downarrow \\
a.\,0 & & 0 \\
\\
\approx & & \approx \\
\\
\tau.\,a.\,0 & \mathcal{R} & 0
\end{array}
$$

- **Chaining** (ie: relational composition) is not respectful
- **What in place of $\approx$?**
  - **Expansion ($\lesssim$)**

    [Arun-Kumarm, Hennessy '91; Milner, Sangiorgi '92]
  - **Controlled relations** [Pous '05]

Example: up-to bisimilarity that fails

$$
\begin{array}{ccc}
\tau.\,a.\,0 & \mathcal{R} & 0 \\
\downarrow & & \Downarrow \\
a.\,0 & & 0 \\
 & & \\
\approx & & \approx \\
 & & \\
\tau.\,a.\,0 & \mathcal{R} & 0
\end{array}
$$

– **Chaining** (ie: relational composition) is not respectful

– **What in place of $\approx$?**

* **Expansion ($\lesssim$)**

  [Arun-Kumarm, Hennessy '91; Milner, Sangiorgi '92]

* **Controlled relations** **[Pous '05]**

- **The culprit: chaining** (ie, relational composition)

- **Everything else: respectfulness is ok**

- **An important use of chaining:**

$$
\begin{array}{ccccc}
P & & \mathcal{R} & & Q \\
\alpha \downarrow & & & & \Downarrow \alpha \\
P' & \approx & \mathcal{F}(\mathcal{R}) & \approx & Q'
\end{array}
$$

- **What should we use in place of $\approx$ above?**
  - **Expansion ($\lesssim$)**

    [Arun-Kumarm, Hennessy '91; Milner, Sangiorgi '92]
  - **Controlled relations** [Pous '05]

Theorem

If $\mathcal{F}$ weakly respectful:

$$
\begin{array}{ccccc}
P & & \mathcal{R} & & Q \\
\alpha \big\downarrow & & & & \big\Downarrow \alpha \\
P' & \gtrsim & \mathcal{F}(\mathcal{R}) & \approx & Q'
\end{array}
$$

implies $\mathcal{R} \subseteq \approx$

Powerful:

candidate relations contain only "normal forms"

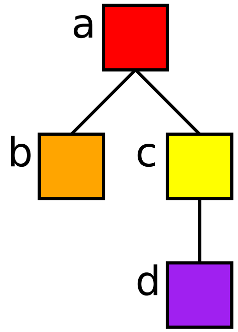
Example: correctness of an abstract machine for (Safe) Ambients

[Giannini, Sangiorgi, Valente, '04]

Nesting of ambients yields a tree

Example:

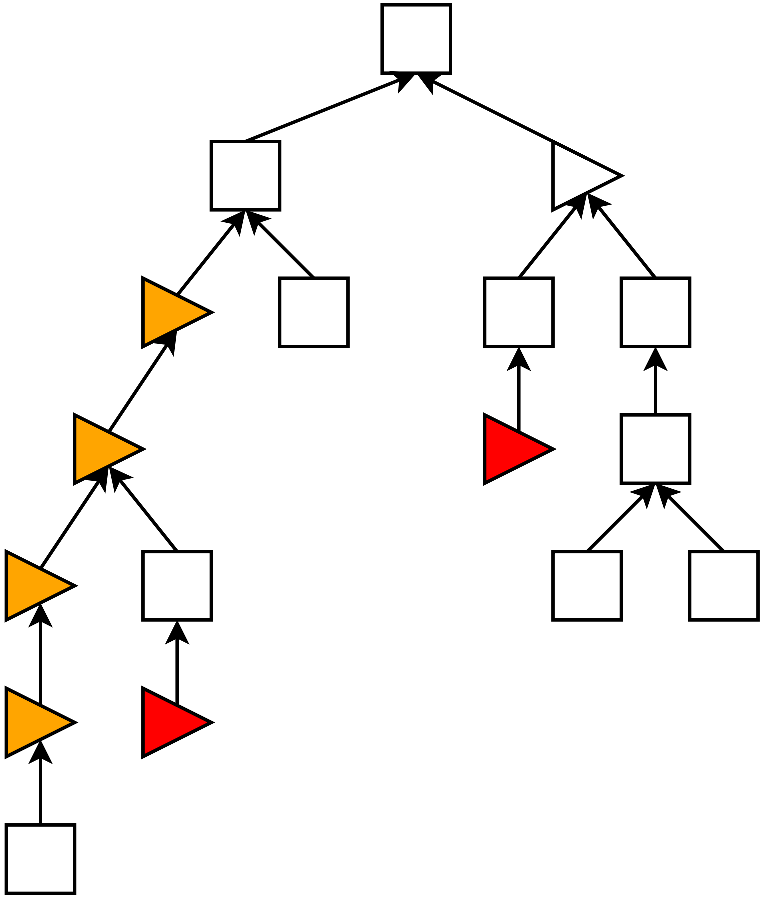
$a\langle b\langle\rangle \mid c\langle d\langle\rangle\rangle\rangle$ becomes



Movements of ambients:

- modify the tree structures
- can produce **forwarders**

The abstract machine – graphical representation



- Forwarders: common in distributed systems
- Forwarder chains
- Possible useless forwarders

Correctness proof (sketch)

– Ideally, using $\mathcal{R} \triangleq \{(\llbracket P \rrbracket, P)\}$

P : an Ambient term

$\llbracket P \rrbracket$: representation of P in the AM

– However:

$$\begin{array}{ccc} \llbracket P \rrbracket & \mathcal{R} & P \\ \downarrow & & \downarrow \\ \nu h\ (h \triangleright k \mid \llbracket Q(h) \rrbracket) & \cancel{\mathcal{R}} & Q(k) \end{array}$$

– Further: in the AM there may be messages floating around (cf: non-atomicity of the implementation of Ambient basic operations)

– Indeed: the bisimulation relation needed is **very** complex.

Expansion ($\lesssim$)

$P \lesssim Q$ if: $\begin{cases} \text{P} \approx \text{Q} \\ \text{P is more efficient than Q} \end{cases}$

Definition:

1. $$\begin{array}{ccc} P & \lesssim & Q \\ \alpha\downarrow & & \Downarrow\alpha \\ P' & \lesssim & Q' \end{array}$$

read: ($\Rightarrow$)

2. $$\begin{array}{ccc} Q & \gtrsim & P \\ \alpha\downarrow & & \downarrow\alpha \\ Q' & \gtrsim & P' \end{array} \qquad \begin{array}{ccc} Q & \gtrsim & P \\ \downarrow & & \| \\ Q' & \gtrsim & P \end{array}$$

read: ($\Rightarrow$)

Examples:

$P \lesssim \tau.P$

$P \not\gtrsim \tau.P$

$\mathcal{R}$ works if we use expansion:

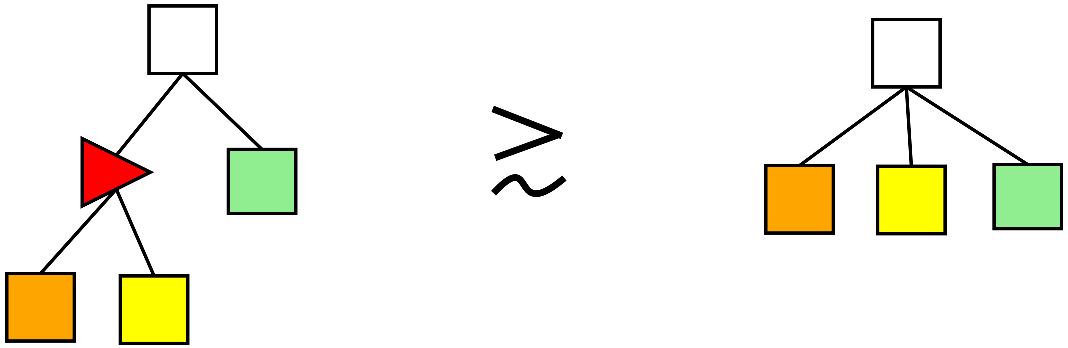
$$
\begin{array}{ccc}
[\![P]\!] & \mathcal{R} & P \\
\downarrow & & \downarrow \\
\nu h\,(h \triangleright k \mid [\![Q(h)]\!]) & & Q(k) \\
\gtrsim & & \approx \\
[\![Q(k)]\!] & \mathcal{R} & Q(k)
\end{array}
$$

Lemma: If h used only for messages in A

$$\boxed{\nu h\,(h \triangleright k \mid A) \gtrsim A\{{}^{k}\!/\!{}_{h}\}}$$

Similarly for features other than forwarders

Lemma $\nu h\,(h \rhd k \mid A) \gtrsim A\{ {}^{k}\!/h \}$



- **Simple proof**
  - local property of the AM
  - up-to techniques applicable to expansion (ex: expansion up-to expansion)

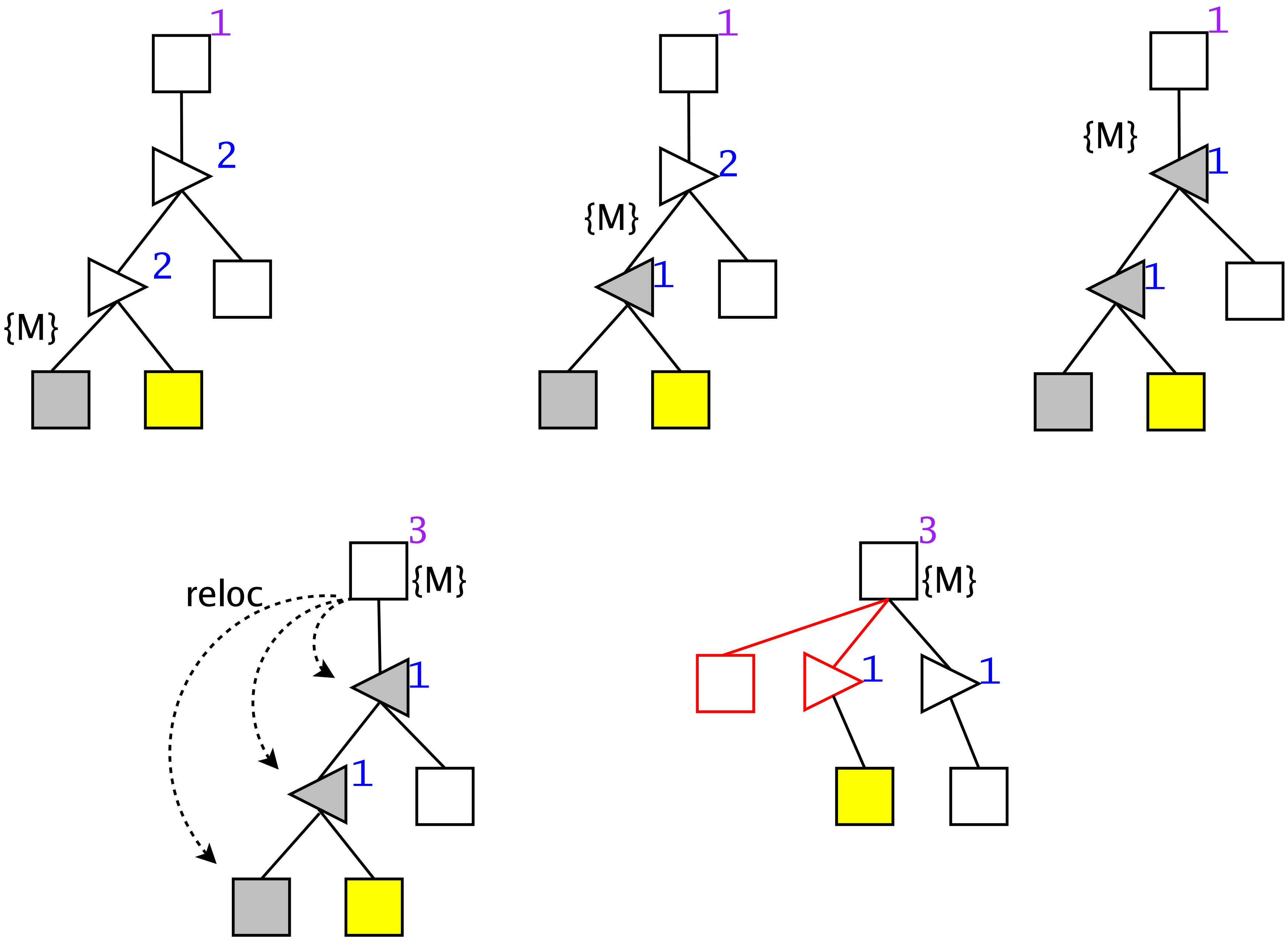
Rigidity of expansion

$P \lesssim Q$ says: P is better at **every** step

Example: optimise the AM **[Hirschkoff, Pous, Sangiorgi, '05]**

- **garbage collection of useless forwarders**
  - use counters in forwarders (= number of children)
- **remove chains of forwarders**
  - adapt Tarjan's union-find algorithm (**relocation**)

Relocation (cf: Tarjan sets)



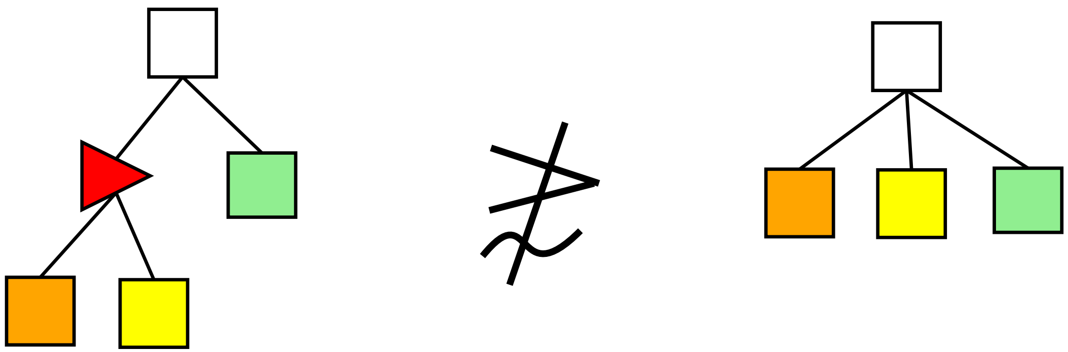
$$\llbracket P \rrbracket \quad \triangleq \quad \text{the original machine}$$
$$[P] \quad \triangleq \quad \text{the optimised machine}$$

– $[P]$ **obviously better, but** $[P] \not\gtrsim \llbracket P \rrbracket$

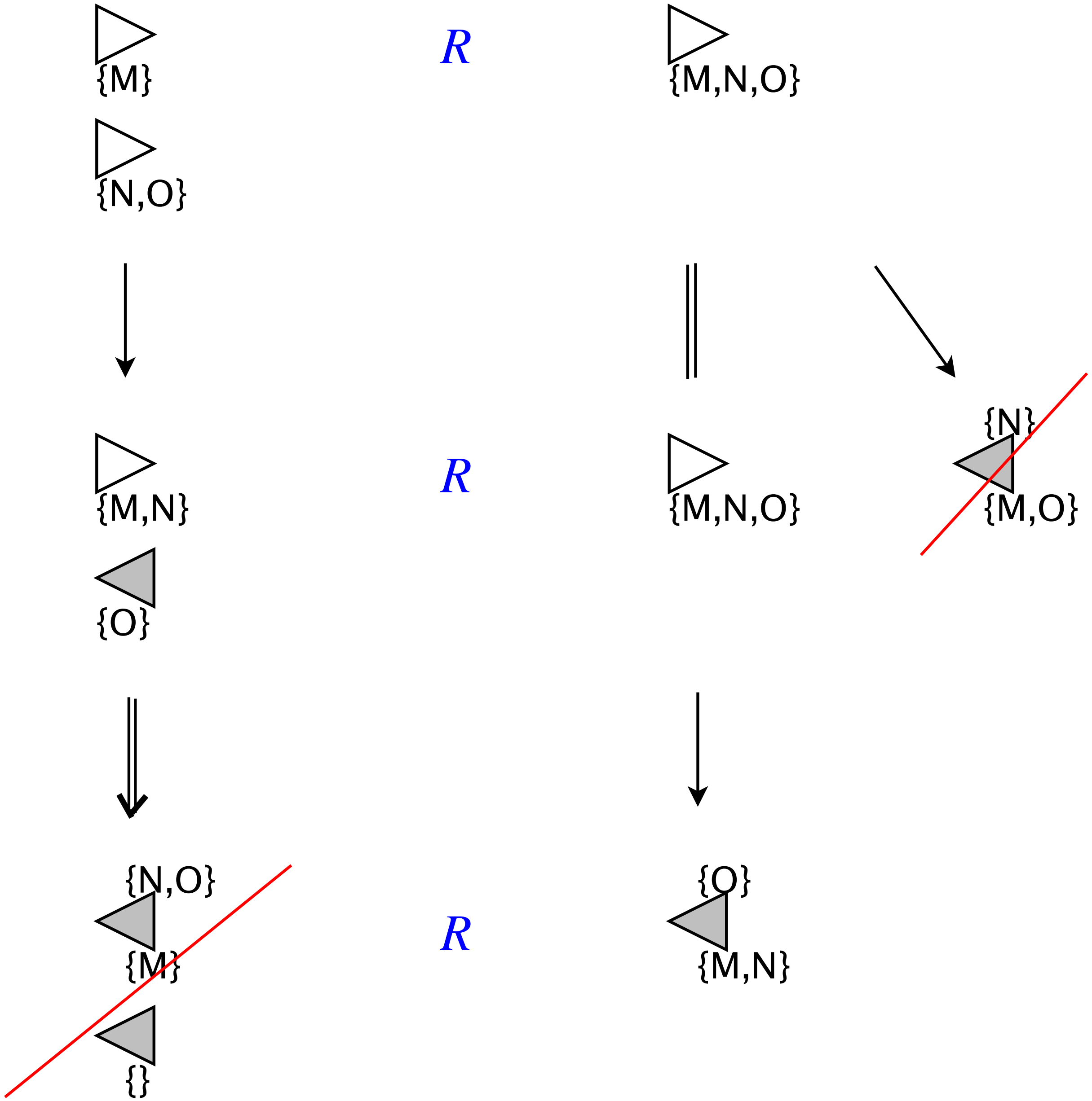
  ∗ initial adminstrative work, that only later pays off

– **Worst: in the optimised machine:**

$$\boxed{\nu h\, (h \rhd k \mid A) \not\gtrsim A\{{}^{k}/h\}}$$



$$\nu h\,(h \triangleright k \mid A) \not\gtrsim A\{{}^{k}/h\}$$



Equivalence between the two machines

– Ideally, using $\mathcal{R} \triangleq \{(\llbracket P \rrbracket, [P])\}$ (normal forms)

  But $\mathcal{R}$ is not a bisimulation up-to expansion

– **Correctness proof in [Hirschkoff, Pous, Sangiorgi, '05]: a full bisimulation**

– **[Pous 05]: A proposal for relations more flexible than $\lesssim$**

  * Now $\mathcal{R}$ works
  * Define properties needed in a relation for the "up-to"

    Example: termination of the transitive closure
  * The relation need not be behaviourally interesting
  * Drawbacks: proving the conditions, compositionality

Conclusions, part IV

- **$\lesssim$, or other relations:**
  - needed to control silent moves
  - allow us to reduce candidate relations only normal forms
- **$\lesssim$: nice mathematical properties, sometimes too rigid**

Problem 4: up-to in the weak case

a) Improvements of $\lesssim$

- better notion of "efficiency"
- more powerful up-to

  (goal: normal forms in candidate relations)

b) Composition of up-to techniques

- how can chaining be replaced?

Important! (practical relevance of **weak** bisimilarity)

Partial results: [Pous 05]

Problem 5: Mechanical verification

- **How can these enhancements be integrated in tools ?**
- **Partial results** **[Hirschkoff]**
  - theorem provers
  - automatic checking
  - Applied to **infinite-state** processes

Problem 6: Other primitive techniques

– Example: up-to substitutions sound in the π-calculus

$$\begin{tikzcd} P \arrow[d, "\alpha"'] & \mathcal{R} & Q \arrow[d, "\alpha"] \\ P'\cancel{\sigma} & \mathcal{R} & Q'\cancel{\sigma} \end{tikzcd} \quad \text{implies } \mathcal{R} \subseteq \sim \text{ ?}$$

References

This course is based on the draft book:

- Davide Sangiorgi, *An introduction to bisimulation and coinduction*, Draft, 2009

Please contact me if you'd like to read and comment parts of it.

Focus lab

A joint initiative between **INRIA (France)** and **Univ. Bologna (Italy)**

Permanent members: M. Bravetti, U. Dal Lago, M. Gabbrielli, C. Laneve, S. Martini, D. Sangiorgi, G. Zavattaro.

Scientific theme: semantic foundations for distributed software systems (ubiquitous systems)

Eg: methods for the analysis and synthesis, at various levels of abstraction; issues of expressiveness

Central concepts: ‘interaction’, ’component’

Basis: logics, types, algebra, operational semantics

NB: A recently established **“Collegio di Cina”** within the University of Bologna