Some remarks on Bisimulation and Coinduction

Davide Sangiorgi

University of Bologna

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

Edinburgh, April 2012

The ’91 Turing Award to

Arthur John Robin Gorell Milner

From `http://amturing.acm.org/`

“For three distinct and complete achievements:

1. LCF
2. ML
3. CCS.

In addition, he formulated and strongly advanced full abstraction”

No bisimulation and coinduction

Another fundamental contribution for Milner: Bisimulation and Coinduction

Bisimulation, bisimilarity, coinduction

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 ($\sim$) :

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

(coind. definition)

Hence:

$$\frac{P \;\mathcal{R}\; Q \qquad \mathcal{R} \text{ is a bisimulation}}{P \sim Q}$$

(coind. proof principle)

Major contributions to concurrency theory...

- To **define equality** on processes (fundamental !!)
- To **prove equalities**
  - ∗ even if bisimilarity is not the chosen equivalence
    - trying bisimilarity first
    - coinductive characterisations of the chosen equivalence
- To **justify algebraic laws**
- To **minimise** the state space
- To **abstract** from certain details

In fact, major contributions to computer science...

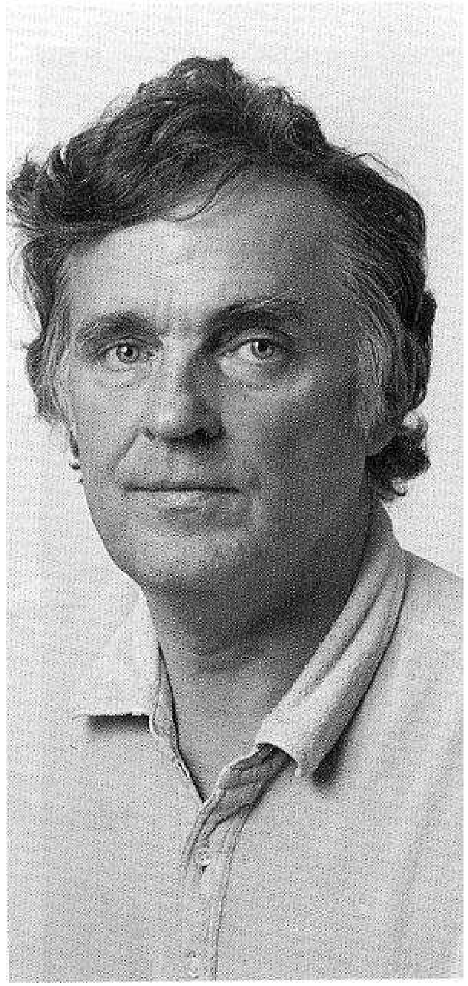
- Functional languages and OO languages
- Program analysis
- Verification tools:
- Type theory
- Databases
- Compiler correctness

And beyond computer science....

- Set Theory and Mathematics
- Modal Logics
- Artificial Intelligence
- Cognitive Science
- Philosophy
- Physics

The discovery of bisimulation and coinduction

Robin Milner



David Park

Milner, early 1970s

Session No. 11 Theoretical Foundations 1971

AN ALGEBRAIC DEFINITION OF SIMULATION BETWEEN PROGRAMS*

Robin Milner

Computer Science Department
Stanford University
Stanford, California

A simulation relation between programs is defined which is a quasi-ordering. Mutual simulation is then an equivalence relation, and by dividing out by it we abstract from a program such details as how the sequencing is controlled and how data is represented. The equivalence classes are approximations to the algorithms which are realized, or expressed, by their member programs.

A technique is given and illustrated for proving simulation and equivalence of programs; there is an analogy with Floyd's technique for proving correctness of programs. Finally, necessary and sufficient conditions for simulation are given.

DESCRIPTIVE TERMS: Simulation, weak homomorphism, algorithm, program correctness, program equivalence.

A formal notion of simulation between programs. Memo 14, Comp. and Logic Research Group, University of Swansea, 1970

Program simulation: an extended formal notion. Memo 17, Comp. and Logic Research Group, University of Swansea, 1971

An algebraic definition of simulation between programs 2nd International Joint Conferences on Artificial Intelligence, London, 1971

- **Programs: partial, sequential, imperative**
- **Program correctness**
- **When 2 programs realise the same algorithm?**
- **Milner's proposal: simulation**
- not quite today's simulation
  the proof technique, locality
- tree-like computation and concurrency mentioned for future work
- ... but Milner never looked into that
  (bisimulation might have been discovered)

Milner, later in the 1970s

A novel theory of processes (**CCS**) where **behavioural equivalence** is fundamental and based on **locality**

$$
\begin{array}{ccc}
P & \sim_{n+1} & Q \\
a \downarrow & & \downarrow a \\
P' & \sim_n & Q'
\end{array}
$$

$$\sim_0 \triangleq \mathcal{P} \times \mathcal{P}$$

$$\sim_\omega \triangleq \bigcap_n \sim_n$$

A Calculus of Communicating Systems LNCS 92, Springer, 1980

Lemma $\sim_\omega$ **is not invariant under transitions**

Park, 80/81: sabbatical in Edinburgh

– **Staying at Milner's (!)**

– **A fixed-point reading of Milner's theory:**

The definition of $\sim_\omega$ is based on a functional $\mathcal{F}$ that is

* $*$ monotone
* $*$ non-cocontinuous

– **Applying fixed-point theory:**

Bisimilarity $(\sim) \triangleq \text{gfp}(\mathcal{F})$

A bisimulation : a post-fixed point of $\mathcal{F}$

Corollary : any bisimulation $\subseteq \sim$

$\sim \triangleq \bigcap_{\boldsymbol{\lambda}\ \text{ordinal}} \mathcal{F}^{\boldsymbol{\lambda}}(\mathcal{P} \times \mathcal{P})$

if you buy a big enough house you can benefit from other people's ideas

— Milner

Milner's insights

- an equivalence based on locality
- the proof technique

And he made popular both bisimulation and coinduction

- CCS
- Milner and Tofte. Co-induction in relational semantics. TCS, 1991, and Tech. Rep. LFCS, Edinburgh, 1988.

Origins of the names

Milner and Park, after the breakfast in which bisimulation came up:

> We went for a walk in the hills in the afternoon, wondering what to call the equivalence. He wanted "mimicry", which I thought a bad idea (it's a hard word to pronounce!). I suggested "bisimulation"; his first reaction was "too many syllables"; I replied that it was easy to pronounce. I won.
>
> — Milner

Coinduction

- Barwise and Etchemendy, “The Liar: an Essay in Truth and Circularity”, 1987
- Milner and Tofte, “Co-induction in relational semantics”. Tech. Rep. LFCS, Edinburgh, 1988.

Why bisimulation and coinduction discovered so late?

Weak homomorphism in automata theory

– well-known in the 1960s

[cf: Ginzburg's book]

– Milner's simulation, algebraically

Algorithm for minimisation of automata

[Huffman 1954 and Moore 1956]

[also: the Myhill-Nerode theorem 1957-58]

Find the **non-equivalent states**, as an inductive set N:

1. If s **final** and t **is not**, then $s\ N\ t$
2. if $\exists\ a$ s.t. $\sigma(s,a)\ N\ \sigma(s,a)$ then $s\ N\ t$

The complement set: the **equivalent states**

What is this complement set?

The largest relation $\mathcal{R}$ s.t.

1. s **final** and $s\ \mathcal{R}\ t$ imply t **final**, and the converse
2. $\forall\ a$, if $s\ \mathcal{R}\ t$ then $\sigma(s, a)\ \mathcal{R}\ \sigma(s, a)$

[cf: bisimilarity]

NB: any relation with 1-2 above relates equivalent states

[cf: bisimulation]

The appearance of bisimulation in Set Theory

Foundations of set theory (cf: non-well-founded sets)

- **Forti, Honsell** '80-83, **Hinnion** '80-81

  Bisimulations: f-conservative relations, contractions

  Coinduction?

  * yes
  * a little hidden (more attention to bisimulation equivalences than bisimulations)

- **Aczel** '85-89

  nwf sets popular, motivated by Milner's work on CCS

  the basis of the coalgebraic approach to semantics

Much earlier than that....

– Dimitry Mirimanoff [1917] (“ensembles extraordinaires”)

Isomorphism between two nwf sets E and E':

A perfect correspondence can be established between the elements of E and E', in such a way that:

1. all atoms $e \in E$ corresponds to an atom $e \in E'$ and conversely;
2. all sets $F \in E$ corresponds to a set $F' \in E'$ so that the perfect correspondence can also be established on F and F' (ie, all atoms in F corresponds to an atom in F', and so forth)

For Mirimanoff: **isomorphism is not equality**

(cf: Zermelo's extensionality axiom)

Hence **isomorphism remains different from bisimilarity**

Example:

$A = \{B\}$ and $B = \{A\}$ isomorphic, not equal

$\{A, B\}$ not isomorphic to $\{A\}$ or $\{B\}$

Had one investigated the impact of isomorphism on extensionality, bisimulation and bisimilarity would have been discovered

We have to wait 65 years : why?

So: why bisimulation has been discovered so late?

– Dangers of circularity and paradoxes (like Burali-Forti's and Russel's)

– Russel's stratified approach

– Common sense

– Lack of concrete motivations

So: why bisimulation has been discovered so late?

- Dangers of circularity and paradoxes (like Burali-Forti's and Russel's)
- Russel's stratified approach
- Common sense
- Lack of concrete motivations
- **none of these entirely convincing (cf: automata theory)**

So: why bisimulation has been discovered so late?

- Dangers of circularity and paradoxes (like Burali-Forti's and Russel's)
- Russel's stratified approach
- Common sense
- Lack of concrete motivations
- **none of these entirely convincing (cf: automata theory)**
- **.... because Robin had not thought about it earlier**

For the future

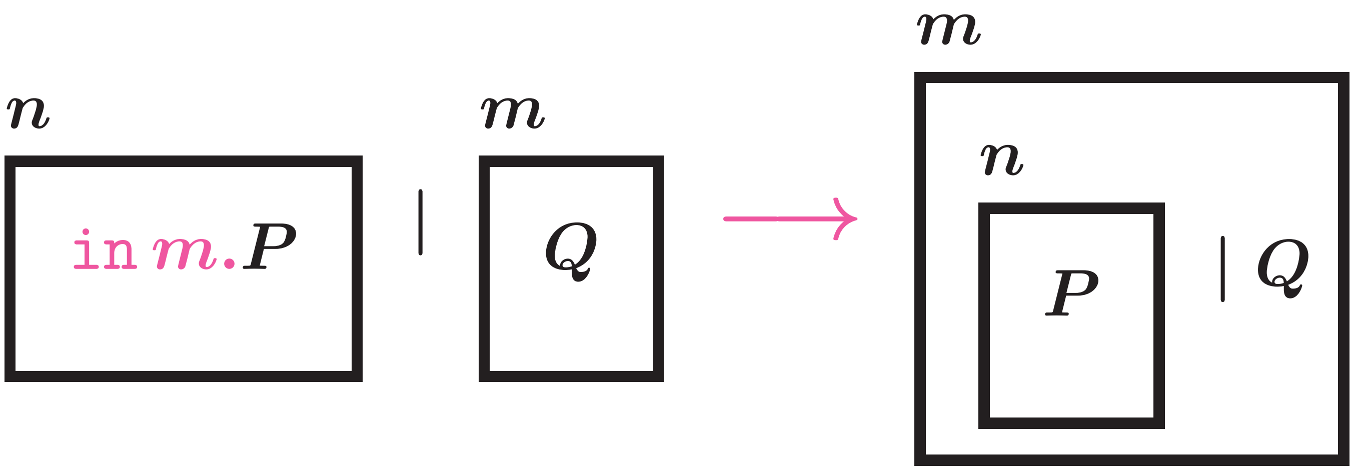
- metatheory
- probabilistic coinduction
- higher-order languages
- ...

Enhancements of the bisimulation/coinduction proof method

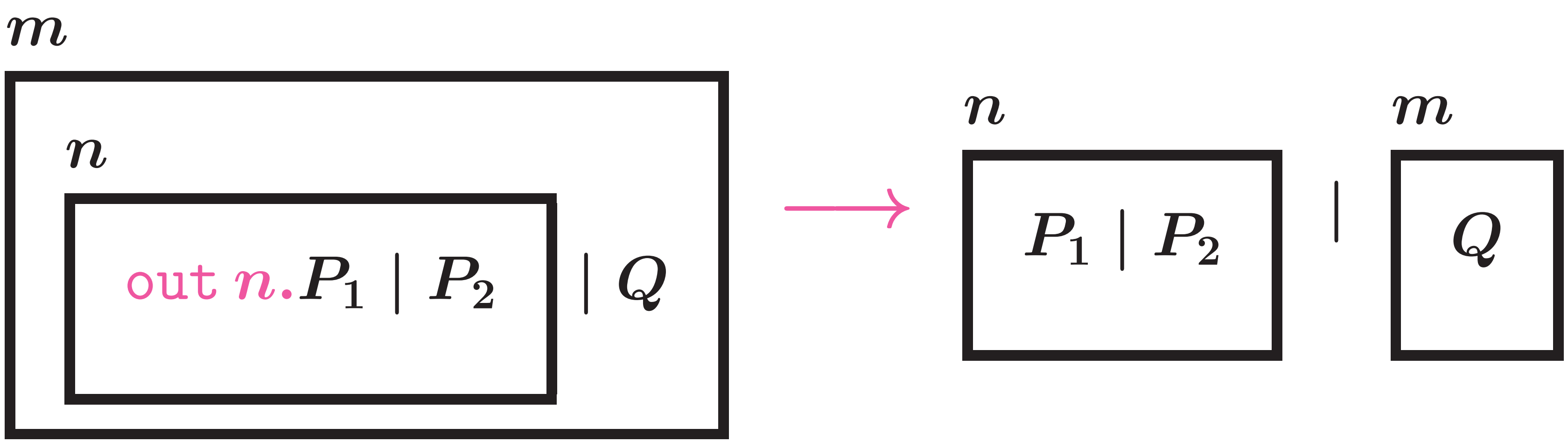
Ambients: syntax

			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



Enhancements of the method: an example

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

(the loop: simplifies the example, not necessary)

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

Try again...

Is this a bisimulation?

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

Is this a bisimulation?

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

No!

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

$$\begin{array}{ccc} k\langle Q \mid \nu n\ \ n\langle P\rangle\ \rangle & \mathcal{R} & k\langle Q\rangle \mid 0 \\ \downarrow & & \downarrow \\ k\langle Q' \mid \nu n\ \ n\langle P\rangle\ \rangle \mid h\langle R\rangle & \not{\mathcal{R}} & k\langle Q'\rangle \mid h\langle R\rangle \mid\ 0 \end{array}$$

Try again...

Is this a bisimulation?

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

Also:

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

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

Try again...

The bisimulation:

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

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

We started with the **singleton** relation

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

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

Can we work with relations smaller than bisimulations?

Advantage: fewer and simpler bisimulation diagrams

Redundant pairs

What we would like to do:

$\mathcal{R} \triangleq \mathcal{R}^* -$ {some redundant pairs}

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ {\scriptstyle\alpha}\downarrow & & \downarrow{\scriptstyle\alpha} \\ P' & \mathcal{R}^* & Q' \end{array} \quad \text{implies } \mathcal{R} \subseteq \sim$$

Redundant pairs

What we would like to do:

$\mathcal{R} \triangleq \mathcal{R}^* - \{$some redundant pairs$\}$

$$\begin{array}{ccc} P & \mathcal{R} & Q \\ \alpha\downarrow & & \downarrow\alpha \\ P' & \mathcal{R}^* & Q' \end{array} \quad \text{implies } \mathcal{R} \subseteq \sim$$

A wrong definition of redundant:

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

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

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

False!

Counterexample

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

$$\begin{array}{lcl} \mathcal{R} & \triangleq & \{(a.\,b, a.\,c)\} \\ R^* & \triangleq & \mathcal{R} \cup \{(b, c)\} \end{array}$$

$$\begin{array}{ccc} a.\,b & \mathcal{R} & a.\,c \\ a\downarrow & & \downarrow a \\ b & \mathcal{R}^* & c \end{array}$$

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

In some cases it works

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

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

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

Warning: in some cases it does not work,
even though $\sim$ is transitive

In some cases it works

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

[Sangiorgi]

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

Warning: in some cases it does not work,
even though the contexts preserve $\sim$

In some cases it works

- 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)**

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

σ: an injective function σ

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

Composition of techniques

diagram :

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

More sophistication $\Rightarrow$

– **more powerful technique**

– **harder soundness proof for the technique**

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

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

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

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

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

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

[Merro, Zappa Nardelli, JACM]

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

(full proof also needs up-to injective substitutions)

Counterexample : up-to context that fails

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

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

Bisimulation is a congruence, yet:

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

Counterexample : up-to context that fails

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

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

Bisimulation is a congruence, yet:

$$\begin{array}{ccccccc}
a.\,0 & & & \mathcal{R} & & & a.\,a.\,0 \\
{\scriptstyle a}\downarrow & & & & & & \downarrow{\scriptstyle a} \\
0 & \sim & \cancel{f}(a.\ 0) & \mathcal{R} & \cancel{f}(a.\ a.\ 0) & \sim & a.\,0
\end{array}$$

Lessons

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

Needed

- **A general theory of enhancements**
  * powerful techniques
  * combination of techniques
  * easy to derive their soundness

  Partial results: [Pous, Sangiorgi]

- **What is a redundant pair?**

  (i.e., a pair for which the bisimulation diagram is not necessary)

- **Robust definition of enhancement**

- **Weak bisimilarity**

  Partial results: [Hirschkoff, Pous]

- **Mechanical verification**

- **Metatheory of bisimulation enhancements**