The Applied Pi Calculus. . . with Proofs

Bruno Blanchet

INRIA Paris-Rocquencourt

joint work with Martín Abadi and Cédric Fournet

April 2015

The applied pi calculus

- Designed by Abadi and Fournet (*Mobile Values, New Names, and Secure Communication*, POPL'01).
- Extension of the pi calculus with terms instead of names for messages.
- Language for modeling security protocols:
  - Terms represent protocol messages.
  - Function symbols represent cryptographic primitives.
  - The properties of these primitives are modeled by equations.
  - The input language of ProVerif is a dialect of the applied pi calculus.
- The applied pi calculus and ProVerif are widely used.
  - Interesting to make them converge, with a solid theoretical foundation.

Our contribution

- Minor changes to the language
    - Closer to ProVerif
- Detailed proofs of all results
    - Minor fixes; some side-conditions were not explicit
    - 74 pages of proofs...
- Revised examples
    - New example on indifferentiability

Related work

- Avik Chaudhuri (private communication, 2007)
  - found a counter-example to "observational equivalence equals labelled bisimilarity", due to a missing side-condition.
- Bengtson et al, LICS'09
  - mentioned a similar counter-example;
  - proposed a framework for defining various extensions of the pi calculus (psi-calculi), with machine-checked proofs.
- Jia Liu (http://lcs.ios.ac.cn/~jliu/papers/LiuJia0608.pdf)
  - made the missing side-condition explicit, and gave a proof of "observational equivalence equals labelled bisimilarity";
  - closer to the original applied pi calculus paper;
  - extension to a stateful variant (POST'14, with Arapinis, Ritter, and Ryan).

Syntax: processes

$L, M, N, T, U, V ::=$	terms
$\quad a, b, c, \ldots, k, \ldots, m, n, \ldots, s$	name
$\quad x, y, z$	variable
$\quad f(M_1, \ldots, M_l)$	function application
$P, Q, R ::=$	processes (or plain processes)
$\quad \mathbf{0}$	null process
$\quad P \mid Q$	parallel composition
$\quad !P$	replication
$\quad \nu n.P$	name restriction ("new")
$\quad \mathit{if}\ M = N\ \mathit{then}\ P\ \mathit{else}\ Q$	conditional
$\quad u(x).P$	message input
$\quad \overline{u}\langle M\rangle.P$	message output

Syntax: processes

$L, M, N, T, U, V ::=$	terms
$\quad a, b, c, \dots, k, \dots, m, n, \dots, s$	$\quad$ name
$\quad x, y, z$	$\quad$ variable
$\quad f(M_1, \dots, M_l)$	$\quad$ function application
$P, Q, R ::=$	processes (or plain processes)
$\quad \mathbf{0}$	$\quad$ null process
$\quad P \mid Q$	$\quad$ parallel composition
$\quad !P$	$\quad$ replication
$\quad \nu n.P$	$\quad$ name restriction ("new")
$\quad \mathit{if}\ M = N\ \mathit{then}\ P\ \mathit{else}\ Q$	$\quad$ conditional
$\quad N(x).P$	$\quad$ message input
$\quad \overline{N}\langle M\rangle.P$	$\quad$ message output

Syntax: extended processes

$A, B, C ::=$	extended processes
P	plain process
$A \mid B$	parallel composition
$\nu n.A$	name restriction
$\nu x.A$	variable restriction
$\{{}^{M}/_{x}\}$	active substitution

- Active substitutions model the knowledge of the adversary.
- $\{{}^{M_1}/_{x_1}, \dots, {}^{M_l}/_{x_l}\}$ for $\{{}^{M_1}/_{x_1}\} \mid \dots \mid \{{}^{M_l}/_{x_l}\}$.
- Substitutions are cycle-free.
- At most one substitution for each variable.
- Exactly one when the variable is restricted.

Sorts

Variables, names, and functions come with sorts:

- $u : \tau$ means that u has sort τ.
  - Examples of sorts: Integer, Key, Data, ...
  - There are infinite numbers of variables and names of each sort.
- $f : \tau_1 \times \cdots \times \tau_l \rightarrow \tau$ means that f has arguments of sorts $\tau_1, \ldots, \tau_l$ and a result of sort τ.

Sorts

Special sort Channel$\langle\tau\rangle$ for channels.

Sorts

Special sort Channel for channels.

- The unsorted applied pi is a particular case of the sorted applied pi, using the single sort Channel.

The sort system enforces that:

- Functional applications are well-sorted.
- M and N are of the same sort in the conditional expression.
- N has sort Channel in the input and output expressions.
    - The sort system can enforce that channels are names or variables: choose types of functions so that no function returns sort Channel.
- Active substitutions preserve sorts.

Semantics: equations

The signature Σ is equipped with an equational theory

- closed under substitutions of terms for variables and names;
  - intuitively, defined from equations that do not contain names;
- respects the sort system;
- non-trivial, that is, there exist two different terms in each sort.

Example

$$
\begin{aligned}
\mathrm{fst}((x, y)) &= x \\
\mathrm{snd}((x, y)) &= y \\
\mathrm{dec}(\mathrm{enc}(x, y), y) &= x \\
\mathrm{check}(x, \mathrm{sign}(x, \mathrm{sk}(y)), \mathrm{pk}(y)) &= \mathrm{ok}
\end{aligned}
$$

Equality modulo the equational theory: $\Sigma \vdash M = N$.

Semantics: preliminary definitions

Processes are considered equal modulo renaming of bound names and variables.

- Needed to define $P\{^M/_x\}$.

A context is a (possibly extended) process with a hole.
An evaluation context is a context whose hole is not under a replication, a conditional, an input, or an output.

$E ::=$	evaluation context
$\_$	hole
$A \mid E$	parallel composition
$E \mid A$	parallel composition
$\nu n.E$	name restriction
$\nu x.E$	variable restriction

Semantics: structural equivalence

Structural equivalence $\equiv$

- equivalence relation
- closed by application of evaluation contexts

$$
\begin{array}{lrcl}
\textsc{Par-}\mathbf{0} & A & \equiv & A \mid \mathbf{0} \\
\textsc{Par-A} & A \mid (B \mid C) & \equiv & (A \mid B) \mid C \\
\textsc{Par-C} & A \mid B & \equiv & B \mid A \\
\textsc{Repl} & !P & \equiv & P \mid !P \\
\\
\textsc{New-}\mathbf{0} & \nu n.\mathbf{0} & \equiv & \mathbf{0} \\
\textsc{New-C} & \nu u.\nu v.A & \equiv & \nu v.\nu u.A \\
\textsc{New-Par} & A \mid \nu u.B & \equiv & \nu u.(A \mid B) \\
 & & & \text{when } u \notin fv(A) \cup fn(A) \\
\\
\textsc{Alias} & \nu x.\{^M/_x\} & \equiv & \mathbf{0} \\
\textsc{Subst} & \{^M/_x\} \mid A & \equiv & \{^M/_x\} \mid A\{^M/_x\} \\
\textsc{Rewrite} & \{^M/_x\} & \equiv & \{^N/_x\} \quad \text{when } \Sigma \vdash M = N
\end{array}
$$

Semantics: internal reduction

Internal reduction $\rightarrow$

- closed by structural equivalence
- closed by application of evaluation contexts

$$
\begin{array}{llcl}
\textsc{Comm} & \overline{N}\langle x\rangle.P \mid N(x).Q & \rightarrow & P \mid Q \\
\textsc{Then} & \mathit{if}\ M = M\ \mathit{then}\ P\ \mathit{else}\ Q & \rightarrow & P \\
\textsc{Else} & \mathit{if}\ M = N\ \mathit{then}\ P\ \mathit{else}\ Q & \rightarrow & Q
\end{array}
$$

for any ground terms M and N
such that $\Sigma \not\vdash M = N$

Using structural equivalence:

$$
\begin{array}{rcll}
\overline{N}\langle M\rangle.P \mid N(x).Q & \equiv & \nu x.(\{^{M}/_{x}\} \mid \overline{N}\langle x\rangle.P \mid N(x).Q) & \\
& \rightarrow & \nu x.(\{^{M}/_{x}\} \mid P \mid Q) & \text{by } \textsc{Comm} \\
& \equiv & P \mid Q\{^{M}/_{x}\} &
\end{array}
$$

Preliminary definitions

- $dom(A)$: domain, set of variables that A exports.
- $fv(A)$: free variables
- A is closed when its free variables are all defined by an active substitution, that is, $dom(A) = fv(A)$.
- $E[\_]$ closes A when $E[A]$ is closed.
- $A \Downarrow a$ when $A \rightarrow^* \equiv E[\overline{a}\langle M\rangle.P]$ for some evaluation context $E[\_]$ that does not bind a.
  - A can send a message on channel a.

Observational equivalence

Definition

An observational bisimulation is a symmetric relation $\mathcal{R}$ between closed extended processes with the same domain such that $A \mathrel{\mathcal{R}} B$ implies:

1. if $A \Downarrow a$, then $B \Downarrow a$;
2. if $A \rightarrow^* A'$ and A' is closed, then $B \rightarrow^* B'$ and $A' \mathrel{\mathcal{R}} B'$ for some B';
3. $E[A] \mathrel{\mathcal{R}} E[B]$ for all closing evaluation contexts $E[\_]$.

Observational equivalence ($\approx$) is the largest such relation.

- Intuitively, $A \approx B$ when an adversary (evaluation context) cannot distinguish A from B.
- Hard to prove because of the universal quantification over all contexts.
  - Use a labeled bisimulation.

Equality in a frame

A frame φ is an extended process built up from $\mathbf{0}$ and active substitutions $\{^M/_x\}$ by parallel composition and restriction.

The frame of A, $\varphi(A)$, is obtained replacing every plain process in A with $\mathbf{0}$.

Definition

Two terms M and N are equal in the frame φ, written $(M = N)\varphi$, if and only if

- $fv(M) \cup fv(N) \subseteq dom(\varphi)$,
- $\varphi \equiv \nu\widetilde{n}.\sigma$, $M\sigma = N\sigma$, and $\{\widetilde{n}\} \cap (fn(M) \cup fn(N)) = \emptyset$ for some names $\widetilde{n}$ and substitution σ.

Independent of the representative $\nu\widetilde{n}.\sigma$.

Static equivalence

Definition

Two closed frames φ and ψ are statically equivalent, written $\varphi \approx_s \psi$, when

- $dom(\varphi) = dom(\psi)$ and
- for all terms M and N, $(M = N)\varphi$ if and only if $(M = N)\psi$.

Two closed extended processes are statically equivalent, written $A \approx_s B$, when their frames are statically equivalent.

- Static equivalence $\varphi \approx_s \psi$ expresses that the frames cannot be distinguished by performing equality tests.
- $A \approx_s B$ expresses that the current knowledge of the adversary in the processes A and B does not allow it to distinguish A from B. The dynamic behavior of A and B is ignored.

Labels

The labelled semantics defines $A \xrightarrow{\alpha} A'$ where α is a label:

- $N(M)$: input of M on channel N;
- $\nu x.\overline{N}\langle x\rangle$: output of x on channel N.
  x must not occur in N.

$bv(N(M)) \stackrel{\text{def}}{=} \emptyset$ and $bv(\nu x.\overline{N}\langle x\rangle) \stackrel{\text{def}}{=} \{x\}$.
$fv(N(M)) \stackrel{\text{def}}{=} fv(N) \cup fv(M)$ and $fv(\nu x.\overline{N}\langle x\rangle) \stackrel{\text{def}}{=} fv(N)$.

The conference paper has labels $\overline{a}\langle u\rangle$ and $\nu u.\overline{a}\langle u\rangle$ for outputs.

- We simplify the semantics by having a single output label.
- One always needs to create a fresh variable for the output message.
- A refined semantics allows $\nu\widetilde{u}.\overline{N}\langle M\rangle$ as label.

Labeled semantics

IN $$N(x).P \xrightarrow{N(M)} P\{^M/_x\}$$

OUT-VAR $$\frac{x \notin fv(\overline{N}\langle M\rangle.P)}{\overline{N}\langle M\rangle.P \xrightarrow{\nu x.\overline{N}\langle x\rangle} P \mid \{^M/_x\}}$$

SCOPE $$\frac{A \xrightarrow{\alpha} A' \qquad u \text{ does not occur in } \alpha}{\nu u.A \xrightarrow{\alpha} \nu u.A'}$$

PAR $$\frac{A \xrightarrow{\alpha} A' \qquad bv(\alpha) \cap fv(B) = \emptyset}{A \mid B \xrightarrow{\alpha} A' \mid B}$$

STRUCT $$\frac{A \equiv B \qquad B \xrightarrow{\alpha} B' \qquad B' \equiv A'}{A \xrightarrow{\alpha} A'}$$

Labeled bisimilarity

Definition

A labelled bisimulation is a symmetric relation $\mathcal{R}$ on closed extended processes such that $A \mathrel{\mathcal{R}} B$ implies:

1. $A \approx_s B$;
2. if $A \rightarrow A'$ and A' is closed, then $B \rightarrow^* B'$ and $A' \mathrel{\mathcal{R}} B'$ for some B';
3. if $A \xrightarrow{\alpha} A'$, A' is closed, and $fv(\alpha) \subseteq dom(A)$, then $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $A' \mathrel{\mathcal{R}} B'$ for some B'.

Labelled bisimilarity ($\approx_l$) is the largest such relation.

- Item 1 guarantees that the adversary cannot distinguish A from B using its current knowledge.
- Items 2 and 3 guarantee that this property is preserved by reduction.

Main theorem

Theorem

Observational equivalence is labelled bisimilarity: $\approx = \approx_l$.

Bengtson et al's counter example

$$A = \nu a.(\{^a/_x\} \mid x(y).\overline{b}\langle M\rangle.\mathbf{0}) \qquad B = \nu a.(\{^a/_x\} \mid \mathbf{0})$$

- A and B are not observationally equivalent
  - The context $\overline{x}\langle N\rangle$ distinguishes them.
- According to the POPL'01 paper:
  - A and B have the same frame and no transitions,
  - so they are labelled bisimilar.
- A possible fix is to require that exported variables must not be of channel type.
- In our semantics,
  - A has a labelled transition $x(N)$,
  - so A and B are not labelled bisimilar.

Motivation

- Structural equivalence complicates the analysis of possible reductions in a process.
- In a process $A \mid B$,
  - substitutions in A may influence the possible reductions in B,
  - and conversely, substitutions in B may influence reductions in A.

Partial normal forms

Partial formal form of an extended process A:

$$\mathrm{pnf}(A) = \nu\widetilde{n}.(\{^{\widetilde{M}}/_{\widetilde{x}}\} \mid P)$$

with $(fv(P) \cup fv(\widetilde{M})) \cap \{\widetilde{x}\} = \emptyset$.

Formally defined by induction on A.

Lemma

$A \equiv \mathrm{pnf}(A)$.

Structural equivalence on plain processes

Structural equivalence $\overset{\diamond}{\equiv}$ on plain processes

- equivalence relation
- closed by application of evaluation contexts

$$
\begin{array}{llcll}
\textsc{Par-}\mathbf{0}' & P \mid \mathbf{0} & \overset{\diamond}{\equiv} & P & \\
\textsc{Par-A}' & P \mid (Q \mid R) & \overset{\diamond}{\equiv} & (P \mid Q) \mid R & \\
\textsc{Par-C}' & P \mid Q & \overset{\diamond}{\equiv} & Q \mid P & \\
\textsc{Repl}' & !P & \overset{\diamond}{\equiv} & P \mid !P & \\
\textsc{New-}\mathbf{0}' & \nu n.\mathbf{0} & \overset{\diamond}{\equiv} & \mathbf{0} & \\
\textsc{New-C}' & \nu n.\nu n'.P & \overset{\diamond}{\equiv} & \nu n'.\nu n.P & \\
\textsc{New-Par}' & P \mid \nu n.Q & \overset{\diamond}{\equiv} & \nu n.(P \mid Q) & \text{when } n \notin fn(P) \\
\textsc{Rewrite}' & P\{^M/_x\} & \overset{\diamond}{\equiv} & P\{^N/_x\} & \text{when } \Sigma \vdash M = N
\end{array}
$$

Structural equivalence on partial normal forms

Structural equivalence $\overset{\circ}{\equiv}$ on extended processes in partial normal form

- equivalence relation

$$\frac{P \overset{\diamond}{\equiv} P' \qquad (fv(P) \cup fv(P')) \cap dom(\sigma) = \emptyset}{\nu\tilde{n}.(\sigma \mid P) \overset{\circ}{\equiv} \nu\tilde{n}.(\sigma \mid P')}$$

$$\frac{\tilde{n}' \text{ is a reordering of } \tilde{n}}{\nu\tilde{n}.(\sigma \mid P) \overset{\circ}{\equiv} \nu\tilde{n}'.(\sigma \mid P)}$$

$$\frac{n' \notin fn(\sigma)}{\nu\tilde{n}.(\sigma \mid \nu n'.P) \overset{\circ}{\equiv} \nu\tilde{n}, n'.(\sigma \mid P)}$$

$$\frac{\begin{array}{c} dom(\sigma) = dom(\sigma') \qquad \Sigma \vdash x\sigma = x\sigma' \text{ for all } x \in dom(\sigma) \\ (fv(x\sigma) \cup fv(x\sigma')) \cap dom(\sigma) = \emptyset \text{ for all } x \in dom(\sigma) \end{array}}{\nu\tilde{n}.(\sigma \mid P) \overset{\circ}{\equiv} \nu\tilde{n}.(\sigma' \mid P)}$$

Links between structural equivalences

Lemma

- *If* $A \equiv B$, *then* $\mathrm{pnf}(A) \stackrel{\circ}{\equiv} \mathrm{pnf}(B)$.
- *If* $P \stackrel{\diamond}{\equiv} Q$, *then* $P \equiv Q$.
- *If* $A \stackrel{\circ}{\equiv} B$, *then* $A \equiv B$.

By induction on the derivations.

Internal reduction

Internal reduction $\rightarrow_\diamond$ on plain processes

- closed by $\stackrel{\diamond}{\equiv}$
- closed by application of evaluation contexts

$$
\begin{array}{lrcl}
\textsc{Comm}' & \overline{N}\langle M\rangle.P \mid N(x).Q & \rightarrow_\diamond & P \mid Q\{^M/_x\} \\
\textsc{Then}' & \mathit{if}\ M = M\ \mathit{then}\ P\ \mathit{else}\ Q & \rightarrow_\diamond & P \\
\textsc{Else}' & \mathit{if}\ M = N\ \mathit{then}\ P\ \mathit{else}\ Q & \rightarrow_\diamond & Q
\end{array}
$$

for any ground terms M and N
such that $\Sigma \not\vdash M = N$

Internal reduction $\rightarrow_\circ$ on extended processes in partial normal form

- closed by $\stackrel{\circ}{\equiv}$
- $\dfrac{P \rightarrow_\diamond P'}{\nu\widetilde{n}.(\sigma \mid P) \rightarrow_\circ \nu\widetilde{n}.(\sigma \mid P')}$

Link between internal reductions

Lemma

- *If* $A \rightarrow B$*, then* $\mathrm{pnf}(A) \rightarrow_\circ \mathrm{pnf}(B)$*.*
- *If* $P \rightarrow_\diamond Q$*, then* $P \rightarrow Q$*.*
- *If* $A \rightarrow_\circ B$*, then* $A \rightarrow B$*.*

By induction on the derivations.

Labelled reduction on plain processes

Labelled reduction $P \xrightarrow{\alpha}_{\diamond} A$ on plain processes

$$\textsc{In}' \qquad N(x).P \xrightarrow{N(M)}_{\diamond} P\{{}^{M}/_{x}\}$$

$$\textsc{Out-Var}' \qquad \frac{x \notin fv(\overline{N}\langle M\rangle.P)}{\overline{N}\langle M\rangle.P \xrightarrow{\nu x.\overline{N}\langle x\rangle}_{\diamond} P \mid \{{}^{M}/_{x}\}}$$

$$\textsc{Scope}' \qquad \frac{P \xrightarrow{\alpha}_{\diamond} A \qquad n \text{ does not occur in } \alpha}{\nu n.P \xrightarrow{\alpha}_{\diamond} \nu n.A}$$

$$\textsc{Par}' \qquad \frac{P \xrightarrow{\alpha}_{\diamond} A \qquad bv(\alpha) \cap fv(Q) = \emptyset}{P \mid Q \xrightarrow{\alpha}_{\diamond} A \mid Q}$$

$$\textsc{Struct}' \qquad \frac{P \overset{\diamond}{\equiv} Q \qquad Q \xrightarrow{\alpha}_{\diamond} B \qquad B \equiv A}{P \xrightarrow{\alpha}_{\diamond} A}$$

Labelled reduction on partial normal forms

Labelled reduction $A \xrightarrow{\alpha}_\circ B$, where

- A is an extended process in partial normal form and
- B is an extended process

$$\frac{\begin{array}{c} A \overset{\circ}{\equiv} \nu\widetilde{n}.(\sigma \mid P) \qquad P \xrightarrow{\alpha'}_\diamond B' \qquad B \equiv \nu\widetilde{n}.(\sigma \mid B') \\ fv(\sigma) \cap bv(\alpha') = \emptyset \qquad \Sigma \vdash \alpha\sigma = \alpha' \\ \text{the elements of } \widetilde{n} \text{ do not occur in } \alpha \end{array}}{A \xrightarrow{\alpha}_\circ B}$$

Links between labelled reductions

Lemma (Characterization of labelled reductions)

$P \xrightarrow{\alpha}_\diamond A$ if and only if for some $\widetilde{n}$, P_1, P_2, A_1, N, M, P', x, $P \overset{\diamond}{\equiv} \nu\widetilde{n}.(P_1 \mid P_2)$, $A \equiv \nu\widetilde{n}.(A_1 \mid P_2)$, $\{\widetilde{n}\} \cap fn(\alpha) = \emptyset$, $bv(\alpha) \cap fv(P_1 \mid P_2) = \emptyset$, and one of the following two cases holds:

1. $\alpha = N(M)$, $P_1 = N(x).P'$, *and* $A_1 = P'\{{}^M/_x\}$; *or*
2. $\alpha = \nu x.\overline{N}\langle x\rangle$, $P_1 = \overline{N}\langle M\rangle.P'$, *and* $A_1 = P' \mid \{{}^M/_x\}$.

Lemma

- *If* $A \xrightarrow{\alpha} B$, *then* $\mathrm{pnf}(A) \xrightarrow{\alpha}_\circ B$.
- *If* $P \xrightarrow{\alpha}_\diamond A$, *then* $P \xrightarrow{\alpha} A$.
- *If* $A \xrightarrow{\alpha}_\circ B$, *then* $A \xrightarrow{\alpha} B$.

Decomposition of labelled reductions: plain processes

Lemma

Suppose that P_0 is closed, α is $\nu x.\overline{N'}\langle x\rangle$ or $N'(M')$ for some ground term N', and $P_0 \xrightarrow{\alpha}_{\diamond} A$. Then one of the following cases holds:

1. $P_0 = P \mid Q$ *and either* $P \xrightarrow{\alpha}_{\diamond} A'$ *and* $A \equiv A' \mid Q$, *or* $Q \xrightarrow{\alpha}_{\diamond} A'$ *and* $A \equiv P \mid A'$, *for some* P, Q, *and* A';
2. $P_0 = \nu n.P$, $P \xrightarrow{\alpha}_{\diamond} A'$, *and* $A \equiv \nu n.A'$ *for some* P, A', *and* n *that does not occur in* α;
3. $P_0 = !P$, $P \xrightarrow{\alpha}_{\diamond} A'$, *and* $A \equiv A' \mid !P$ *for some* P *and* A';
4. $P_0 = N(x).P$, $\alpha = N'(M')$, $\Sigma \vdash N = N'$, *and* $A \equiv P\{^{M'}/_{x}\}$ *for some* N, x, P, N', *and* M';
5. $P_0 = \overline{N}\langle M\rangle.P$, $\alpha = \nu x.\overline{N'}\langle x\rangle$, $\Sigma \vdash N = N'$, $x \notin fv(P_0)$, *and* $A \equiv P \mid \{^{M}/_{x}\}$ *for some* N, M, P, x, *and* N'.

Decomposition of labelled reductions: partial normal forms

Lemma

If

- $\nu\widetilde{n}.(\sigma \mid P)$ *is a closed extended process in partial normal form,*
- $\nu\widetilde{n}.(\sigma \mid P) \xrightarrow{\alpha}_{\circ} A$,
- $fv(\alpha) \subseteq dom(\sigma)$, *and*
- *the elements of* $\widetilde{n}$ *do not occur in* α,

then $P \xrightarrow{\alpha\sigma}_{\diamond} A'$, $A \equiv \nu\widetilde{n}.(\sigma \mid A')$, *and* $bv(\alpha) \cap dom(\sigma) = \emptyset$ *for some* A'.

Composition of labelled reductions

Lemma

If

- P *and* Q *are closed processes,* N *is a ground term,*
- $P \xrightarrow{N(x)}_\diamond A$*, and*
- $Q \xrightarrow{\nu x.\overline{N}\langle x\rangle}_\diamond B$*,*

then $P \mid Q \rightarrow_\diamond R$ *and* $R \equiv \nu x.(A \mid B)$ *for some* R*.*

Decomposition of internal reductions: plain processes

Lemma

Suppose that P_0 is a closed process and $P_0 \rightarrow_\diamond R$. Then one of the following cases holds:

1. $P_0 = P \mid Q$ *for some* P *and* Q, *and one of the following cases holds:*
   1. $P \rightarrow_\diamond P'$ *and* $R \equiv P' \mid Q$ *for some closed process* P',
   2. $P \xrightarrow{N(x)}_\diamond A$, $Q \xrightarrow{\nu x.\overline{N}\langle x\rangle}_\diamond B$, *and* $R \equiv \nu x.(A \mid B)$ *for some* A, B, x, *and ground term* N,

   and two symmetric cases obtained by swapping P *and* Q;
2. $P_0 = \nu n.P$, $P \rightarrow_\diamond Q'$, *and* $R \equiv \nu n.Q'$ *for some* n *and some closed processes* P *and* Q';
3. $P_0 = !P$, $P \mid P \rightarrow_\diamond Q'$, *and* $R \equiv Q' \mid !P$ *for some closed processes* P *and* Q'.
4. $P_0 =$ *if* $M = N$ *then* P *else* Q *and either* $\Sigma \vdash M = N$ *and* $R \equiv P$, *or* $\Sigma \vdash M \neq N$ *and* $R \equiv Q$, *for some* M, N, P, *and* Q.

Decomposition of internal reductions: partial normal forms

Lemma

If

- $\nu\widetilde{n}.(\sigma \mid P)$ *is a closed extended process in partial normal form and*
- $\nu\widetilde{n}.(\sigma \mid P) \rightarrow_{\circ} A$,

then $P \rightarrow_{\diamond} P'$ *and* $A \equiv \nu\widetilde{n}.(\sigma \mid P')$ *for some closed process* P'.

Proof technique

- Induction on the derivations.
- Strenghten the inductive invariant, to be able to apply the current lemma to a derivation built as a result of applying the inductive hypothesis.

Static equivalence

Lemma

Static equivalence is

- *invariant by structural equivalence and reduction, and*
- *closed by application of closing evaluation contexts.*

For the second point,

- show that we can restrict ourselves to contexts $E = \nu\widetilde{u}.(\_ \mid C)$ such that all subcontexts of E are closing.
- proceed by structural induction on E.

Context closure

Lemma

$\approx_l$ is closed by application of closing evaluation contexts.

- Restrict attention to contexts of the form $\nu\widetilde{u}.(\_ \mid C)$.
- To every relation $\mathcal{R}$ on closed extended processes, we associate $\mathcal{R}' = \{(\nu\widetilde{u}.(A \mid C), \nu\widetilde{u}.(B \mid C)) \mid A \mathrel{\mathcal{R}} B, \nu\widetilde{u}.(\_ \mid C)$ closing for A and $B\}$.
- We prove that, if $\mathcal{R}$ is a labelled bisimulation, then $\mathcal{R}'$ is a labelled bisimulation up to $\equiv$, hence $\mathcal{R} \subseteq \mathcal{R}' \subseteq \approx_l$.
- For $\mathcal{R} = \approx_l$, this property entails that $\approx_l$ is closed by application of evaluation contexts $\nu\widetilde{u}.(\_ \mid C)$.

Context closure

- To every relation $\mathcal{R}$ on closed extended processes, we associate $\mathcal{R}' = \{(\nu\widetilde{u}.(A \mid C), \nu\widetilde{u}.(B \mid C)) \mid A \mathrel{\mathcal{R}} B, \nu\widetilde{u}.(\_ \mid C) \text{ closing for } A \text{ and } B\}$.
- We prove that, if $\mathcal{R}$ is a labelled bisimulation, then $\mathcal{R}'$ is a labelled bisimulation up to $\equiv$.

Definition

A relation $\mathcal{R}$ on closed extended processes is a labelled bisimulation up to $\equiv$ if and only if $\mathcal{R}$ is symmetric and $A \mathrel{\mathcal{R}} B$ implies:

1. $A \approx_s B$;
2. if $A \rightarrow A'$ and A' is closed, then $B \rightarrow^* B'$ and $A' \equiv\mathcal{R}\equiv B'$ for some closed B';
3. if $A \xrightarrow{\alpha} A'$, A' is closed, and $fv(\alpha) \subseteq dom(A)$, then $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $A' \equiv\mathcal{R}\equiv B'$ for some closed B'.

Context closure

- To every relation $\mathcal{R}$ on closed extended processes, we associate $\mathcal{R}' = \{(\nu\widetilde{u}.(A \mid C), \nu\widetilde{u}.(B \mid C)) \mid A \mathcal{R} B, \nu\widetilde{u}.(\_ \mid C)$ closing for A and $B\}$.
- We prove that, if $\mathcal{R}$ is a labelled bisimulation, then $\mathcal{R}'$ is a labelled bisimulation up to $\equiv$.
  Assume $S \mathcal{R}' T$, with $S = \nu\widetilde{u}.(A \mid C)$, $T = \nu\widetilde{u}.(B \mid C)$, and $A \mathcal{R} B$.
  - $S \approx_s T$ follows from $A \approx_s B$ by a previous lemma.
  - For reductions, consider the partial normal forms of A, B, C: $\mathrm{pnf}(A) = \nu\widetilde{n}.(\sigma \mid P)$, $\mathrm{pnf}(B) = \nu\widetilde{n}'.(\sigma' \mid P')$, $\mathrm{pnf}(C) = \nu\widetilde{n}''.(\sigma'' \mid P'')$.

    A reduction on $S = \nu\widetilde{u}.(A \mid C)$ implies a reduction on $P \mid P''\sigma$, so a reduction on P and/or $P''\sigma$ (by the decomposition lemmas).

    A reduction on P implies a reduction A, so the same reduction on B since $\mathcal{R}$ is a labelled bisimulation, so a reduction on P'.

    A reduction on $P''\sigma$ implies a reduction on $P''\sigma'$ by static equivalence $A \approx_s B$.

    Hence we obtain a reduction on $P' \mid P''\sigma'$, hence on $T = \nu\widetilde{u}.(B \mid C)$.

Characterizing barbs

Lemma

Let A be a closed extended process.

$A \Downarrow a$ if and only if $A \rightarrow^* \xrightarrow{\nu x.\overline{a}\langle x\rangle} A'$ for some fresh variable x and some A'.

$A \equiv E[\overline{a}\langle M\rangle.P]$ for some evaluation context $E[\_]$ that does not bind a if and only if

$A \xrightarrow{\nu x.\overline{a}\langle x\rangle} A'$ for some fresh variable x and some A'.

Labelled bisimilarity implies observational equivalence

Lemma

$\approx_l \subseteq \approx$.

$\approx_l$ satisfies the three properties of observational bisimulations:

1. $\approx_l$ preserves barbs, by characterization of barbs and Properties 2 and 3 of a labelled bisimilation.
2. Suppose that $A \approx_l B$, $A \rightarrow^* A'$, and A' is closed. Close all intermediate processes in $A \rightarrow^* A'$, then conclude that $B \rightarrow^* B'$ and $A' \approx_l B'$ for some B' by Property 2 of a labelled bisimilation.
3. $\approx_l$ is closed by application of closing evaluation contexts, as shown previously.

Moreover, $\approx_l$ is symmetric. Since $\approx$ is the largest observational bisimulation, we obtain $\approx_l \subseteq \approx$.

Observational equivalence implies static equivalence

Lemma

$\approx \subseteq \approx_s$.

If A and B are observationally equivalent, then $A \mid C$ and $B \mid C$ have the same barb $\Downarrow a$ for every $C = \mathit{if}\ \ M = N\ \mathit{then}\ \overline{a}\langle s\rangle$, where a does not occur in A or B and $\mathit{fv}(M) \cup \mathit{fv}(N) \subseteq \mathit{dom}(A)$.

Assuming that A is closed, $\mathit{fv}(M) \cup \mathit{fv}(N) \subseteq \mathit{dom}(A)$, and a does not occur in A, we have
$(M = N)\varphi(A)$ if and only if $A \mid \mathit{if}\ \ M = N\ \mathit{then}\ \overline{a}\langle s\rangle \Downarrow a$.

(Shown using partial normal forms.)

Characterizing inputs

Let $T^p_{N(M)} \stackrel{\text{def}}{=} \overline{p}\langle p\rangle \mid \overline{N}\langle M\rangle.p(x)$.

Lemma

Let A be a closed extended process. Let N and M be terms such that $fv(\overline{N}\langle M\rangle) \subseteq dom(A)$ and p does not occur in A, M, and N.

- *If $A \xrightarrow{N(M)} A'$ and p does not occur in A', then $A \mid T^p_{N(M)} \rightarrow\rightarrow A'$ and $A' \not\Downarrow p$.*
- *If $A \mid T^p_{N(M)} \rightarrow^* A'$ and $A' \not\Downarrow p$, then $A \rightarrow^* \xrightarrow{N(M)} \rightarrow^* A'$.*

Shown using partial normal forms.

Characterizing outputs

Let $T^{p,q}_{\nu x.\overline{N}\langle x\rangle} \stackrel{\text{def}}{=} \overline{p}\langle p\rangle \mid N(x).p(y).\overline{q}\langle x\rangle$.

Lemma

Let A be a closed extended process and N such that $fv(N) \subseteq dom(A)$.

- *If $A \xrightarrow{\nu x.\overline{N}\langle x\rangle} A'$ and p and q do not occur in A, A', and N, then $A \mid T^{p,q}_{\nu x.\overline{N}\langle x\rangle} \rightarrow\rightarrow \nu x.(A' \mid \overline{q}\langle x\rangle)$, $\nu x.(A' \mid \overline{q}\langle x\rangle) \not\Downarrow p$, and $x \notin dom(A)$.*
- *If $A \mid T^{p,q}_{\nu x.\overline{N}\langle x\rangle} \rightarrow^* A''$, $A'' \not\Downarrow p$, $x \notin dom(A)$, and p and q do not occur in A and N, then $A \rightarrow^* \xrightarrow{\nu x.\overline{N}\langle x\rangle} \rightarrow^* A'$ and $A'' \equiv \nu x.(A' \mid \overline{q}\langle x\rangle)$ for some A'.*

Shown using partial normal forms.

Extrusion

Lemma (Extrusion)

Let A and B two closed extended processes with a same domain that contains $\widetilde{x}$. Let $E_{\widetilde{x}}[\_] \stackrel{def}{=} \nu\widetilde{x}.(\prod_{x\in\widetilde{x}} \overline{n_x}\langle x\rangle \mid \_)$ using names n_x that do not occur in A or B. If $E_{\widetilde{x}}[A] \approx E_{\widetilde{x}}[B]$, then $A \approx B$.

If A is a closed extended process with $\{\widetilde{x}\} \subseteq dom(A)$ and $E_{\widetilde{x}}[A] \rightarrow C'$, then $A \rightarrow A'$ and $C' \equiv E_{\widetilde{x}}[A']$ for some closed extended process A'. (Proved using partial normal forms.)

Let $A \mathcal{R} B$ if and only if $\{\widetilde{x}\} \subseteq dom(A) = dom(B)$ and $E_{\widetilde{x}}[A] \approx E_{\widetilde{x}}[B]$, for some $\widetilde{x}$ and some names $\widetilde{n}_x$ that do not occur in A or B.

We show that $\mathcal{R}$ is an observational bisimulation.

Observational equivalence implies labelled bisimilarity

Lemma

$\approx \subseteq \approx_l$.

The relation $\approx$ is symmetric. It satisfies the three properties of labelled bisimulations:

1. If $A \approx B$, then $A \approx_s B$, shown previously.
2. If $A \approx B$, $A \rightarrow A'$, and A' is closed, then $B \rightarrow^* B'$ and $A' \approx B'$ for some B', by Property 2 of the definition of observational bisimulation.
3. If $A \approx B$, $A \xrightarrow{\alpha} A'$, A' is closed, and $fv(\alpha) \subseteq dom(A)$, then $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $A' \approx B'$ for some B'. To prove this property, we rely on characteristic parallel contexts T_α, shown in previous lemmas. In the output case, we obtain a pair $\nu x.(A' \mid \overline{q}\langle x \rangle) \approx \nu x.(B' \mid \overline{q}\langle x \rangle)$, and conclude by the extrusion lemma.

Hence $\approx$ is a labelled bisimulation, and $\approx \subseteq \approx_l$, since $\approx_l$ is the largest labelled bisimulation.

Conclusion

- Importance of detailed proofs.
  - Could be interesting to formalize in a theorem prover, e.g. Coq.
- Partial normal forms are likely to be useful for proving many other results about the applied pi calculus.
- With the minor changes we made, one should be able to show that
  - The plain processes of the applied pi calculus are a subset of the ProVerif input language.
  - The semantics and the notions of observational equivalence match.
- Does anybody want to read the draft?