On the expressivity of the Mobile Ambients and the π-calculus

Linda Brodo

University of Sassari - Italy

AMAST, 23-25 June 2010 - Quebec city

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*;

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;
- **G. Ciobanu, V.A. Zakharov, 2003. Encoding Mobile Ambients into the π-Calculus**;

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;
- **G. Ciobanu, V.A. Zakharov, 2003. Encoding Mobile Ambients into the π-Calculus**;

[D. Gorla, 2009] proves that a translation from MA to π-calculus is impossible. A translation must have:

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;
- **G. Ciobanu, V.A. Zakharov, 2003. Encoding Mobile Ambients into the π-Calculus**;

[D. Gorla, 2009] proves that a translation from MA to π-calculus is impossible. A translation must have:

compositionality, name invariance,

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;
- **G. Ciobanu, V.A. Zakharov, 2003. Encoding Mobile Ambients into the π-Calculus**;

[D. Gorla, 2009] proves that a translation from MA to π-calculus is impossible. A translation must have:

compositionality,	name invariance,
operational correspondance,	divergence reflection,

Previous results on this topic

- L. Cardelli, A. Gordon, 2000. *Mobile Ambients*; (presents a translation from π-calculus to MA)
- D. Sangiorgi, F. Levi, 2000. *Controlling Interference in Ambients*;
- C. Fournet, J.-J. Lévy, A. Schmitt, 2000. *An Asynchronous, Distributed Implementation of Mobile Ambients*;
- **G. Ciobanu, V.A. Zakharov, 2003. Encoding Mobile Ambients into the π-Calculus**;

[D. Gorla, 2009] proves that a translation from MA to π-calculus is impossible. A translation must have:

compositionality,	name invariance,
operational correspondance,	divergence reflection,
success sensitiveness.	

The motivation for this work

A definition of an encoding that clearly shows the distance between the two calculi.

The motivation for this work

A definition of an encoding that clearly shows the distance between the two calculi.
With respect to the requirements in (Gorla, 2009),
we do not have

- divergence reflection

The motivation for this work

A definition of an encoding that clearly shows the distance between the two calculi.
With respect to the requirements in (Gorla, 2009),
we do not have

- divergence reflection

Divergence seems to be the intrinsic distance between MA and π.

The syntax of Mobile Ambients

We consider the pure Mobile Ambients, with recursion.

$$P ::= \mathbf{0} \mid X \mid M.P \mid (\nu n)P \mid P|P' \mid \mu X.P \mid n[P]$$
$$M ::= in\ n \mid out\ n \mid open\ n$$

The syntax of Mobile Ambients

We consider the pure Mobile Ambients, with recursion.

$$P ::= \mathbf{0} \mid X \mid M.P \mid (\nu n)P \mid P|P' \mid \mu X.P \mid n[P]$$
$$M ::= in\ n \mid out\ n \mid open\ n$$

$$\mu X.P \equiv P[\mu X.P/X], \text{up to } \alpha\text{-conversion}$$

The operational semantics of Mobile Ambients

$$n[in\ m.P|Q]|m[R] \rightarrow m[n[P|Q]|R] \qquad In$$

The operational semantics of Mobile Ambients

$$
\begin{array}{ll}
n[in\ m.P|Q]|m[R] \rightarrow m[n[P|Q]|R] & In \\
m[n[out\ m.P|Q]|R] \rightarrow n[P|Q]|m[R] & Out
\end{array}
$$

The operational semantics of Mobile Ambients

$n[in\ m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	*In*
$m[n[out\ m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	*Out*
$open\ n.P \mid n[Q] \rightarrow P \mid Q$	*Open*

The operational semantics of Mobile Ambients

$n[in\ m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	*In*
$m[n[out\ m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	*Out*
$open\ n.P \mid n[Q] \rightarrow P \mid Q$	*Open*
$P \rightarrow Q \Rightarrow (\nu\ n)P \rightarrow (\nu\ n)Q$	*Res*

The operational semantics of Mobile Ambients

$n[in\ m.P \vert Q] \vert m[R] \rightarrow m[n[P \vert Q] \vert R]$	*In*
$m[n[out\ m.P \vert Q] \vert R] \rightarrow n[P \vert Q] \vert m[R]$	*Out*
$open\ n.P \vert n[Q] \rightarrow P \vert Q$	*Open*
$P \rightarrow Q \Rightarrow (\nu\ n)P \rightarrow (\nu\ n)Q$	*Res*
$P \rightarrow Q \Rightarrow n[P] \rightarrow n[Q]$	*Amb*

The operational semantics of Mobile Ambients

$n[in\ m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	*In*
$m[n[out\ m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	*Out*
$open\ n.P \mid n[Q] \rightarrow P \mid Q$	*Open*
$P \rightarrow Q \Rightarrow (\nu\ n)P \rightarrow (\nu\ n)Q$	*Res*
$P \rightarrow Q \Rightarrow n[P] \rightarrow n[Q]$	*Amb*
$P \rightarrow Q \Rightarrow P \mid R \rightarrow Q \mid R$	*Par*

The operational semantics of Mobile Ambients

$n[in\ m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	*In*
$m[n[out\ m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	*Out*
$open\ n.P \mid n[Q] \rightarrow P \mid Q$	*Open*
$P \rightarrow Q \Rightarrow (\nu\ n)P \rightarrow (\nu\ n)Q$	*Res*
$P \rightarrow Q \Rightarrow n[P] \rightarrow n[Q]$	*Amb*
$P \rightarrow Q \Rightarrow P \mid R \rightarrow Q \mid R$	*Par*
$P' \equiv P, P \rightarrow Q, Q \equiv Q' \Rightarrow P' \rightarrow Q'$	$\equiv$

The syntax of π-calculus

A polyadic version, with process identifiers:

$$Q \quad ::= \quad \mathbf{0} \;\mid\; \pi.Q \;\mid\; (\nu a)Q \;\mid\; Q|Q' \;\mid\; Q + Q' \;\mid$$
$$[x = a]Q \;\mid\; [x \neq a]Q \;\mid\; Ide(\tilde{x}) = Q, where\ fn(Q) = \tilde{x}$$

The syntax of π-calculus

A polyadic version, with process identifiers:

$$Q \quad ::= \quad \mathbf{0} \;\mid\; \pi.Q \;\mid\; (\nu a)Q \;\mid\; Q|Q' \;\mid\; Q + Q' \;\mid$$

$$[x = a]Q \;\mid\; [x \neq a]Q \;\mid\; Ide(\tilde{x}) = Q, where\ fn(Q) = \tilde{x}$$

$$\pi \quad ::= \quad a(\tilde{x}) \;\mid\; \overline{a}\ \tilde{b}$$

The operational semantics (just a few rules) of π-calculus

$$\frac{}{\pi.Q \xrightarrow{\pi} Q}\ (prefix)$$

The operational semantics (just a few rules) of π-calculus

$$\frac{}{\pi.Q \xrightarrow{\pi} Q}\ (prefix) \qquad \frac{Q_1 \xrightarrow{\theta} Q_1'}{Q_1 + Q_2 \xrightarrow{\theta} Q_1'}\ (sum)$$

The operational semantics (just a few rules) of π-calculus

$$\frac{}{\pi.Q \xrightarrow{\pi} Q}\ (prefix) \qquad \frac{Q_1 \xrightarrow{\theta} Q_1'}{Q_1 + Q_2 \xrightarrow{\theta} Q_1'}\ (sum)$$

$$\frac{P \xrightarrow{\overline{a}\tilde{b}} P', \{\tilde{b}\} \cap fn(P) = \emptyset}{(\nu\ \tilde{b})P \xrightarrow{\overline{a}\langle\tilde{b}\rangle} P'}\ (bound\ out)$$

The operational semantics (just a few rules) of π-calculus

$$\frac{}{\pi.Q \xrightarrow{\pi} Q}\ (prefix) \qquad \frac{Q_1 \xrightarrow{\theta} Q_1'}{Q_1 + Q_2 \xrightarrow{\theta} Q_1'}\ (sum)$$

$$\frac{P \xrightarrow{\overline{a}\tilde{b}} P', \{\tilde{b}\} \cap fn(P) = \emptyset}{(\nu\ \tilde{b})P \xrightarrow{\overline{a}\langle\tilde{b}\rangle} P'}\ (bound\ out)$$

$$\frac{P \xrightarrow{\overline{a}\langle\tilde{b}\rangle} P', Q \xrightarrow{a(\tilde{x})} Q'}{P|Q \xrightarrow{(\overline{a}\langle\tilde{b}\rangle, a(x))} (\nu\ \tilde{b})(P'|Q'[\tilde{b}/\tilde{x}])}\ (bound\ com)$$

Main mechanisms: who I am, where I am

(a,f) form a pair of unique names, i.e. the coordinates associated to the ambient:

Main mechanisms: who I am, where I am

(a,f) form a pair of unique names, i.e. the coordinates associated to the ambient:

- **a** records the information *who I am* - is the unique name of the ambient,

Main mechanisms: who I am, where I am

(a,f) form a pair of unique names, i.e. the coordinates associated to the ambient:

- **a** records the information *who I am* - is the unique name of the ambient,
- **f** records the information *where I am* - is the unique name of the *father* ambient.

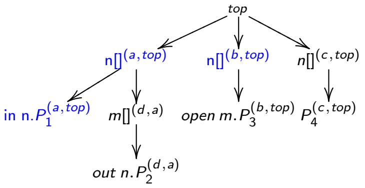
Main mechanisms: who I am, where I am

(a,f) form a pair of unique names, i.e. the coordinates associated to the ambient:

- **a** records the information *who I am* - is the unique name of the ambient,
- **f** records the information *where I am* - is the unique name of the *father* ambient.
- **top** identifies the topmost ambient

The basic dynamics

$A = n[\text{in } n.P_1 \,|\, m[out\ n.P_2]] \mid n[open\ m.P_3] \mid n[P_4]$

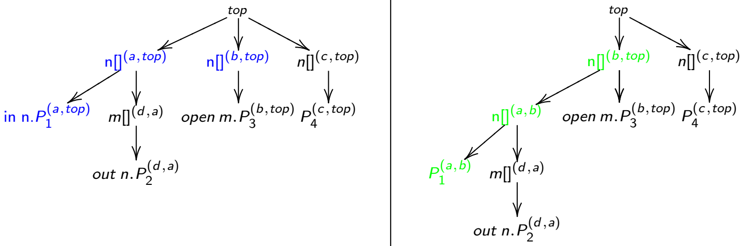


(1)

The basic dynamics

$A = n[\mathsf{in}\ \mathsf{n}.P_1 \,|\, m[out\ n.P_2]] \mid \mathsf{n}[open\ m.P_3] \mid n[P_4]$

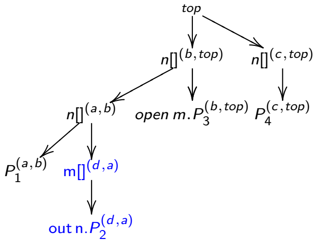
$B = n[n[P_1 \,|\, m[out\ n.P_2]] \mid open\ m.P_3] \mid n[P_4]$



(1) (2)

The basic dynamics

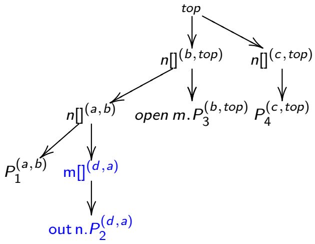
$B = n[n[P_1 \mid m[\text{out n}.P_2]] \mid open\ m.P_3] \mid n[P_4]$



(2)

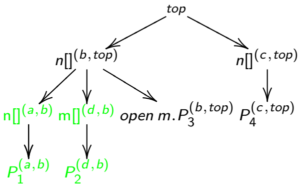
The basic dynamics

$$B = n[n[P_1 \,|\, m[\mathsf{out}\; \mathsf{n}.P_2]] \mid open\; m.P_3] \mid n[P_4]$$



(2)

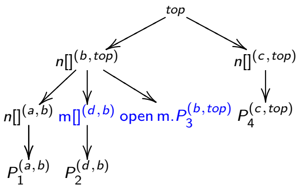
$$C = n[n[P_1]|m[P_2]|open\; m.P_3] \mid n[P_4]$$



(3)

The basic dynamics

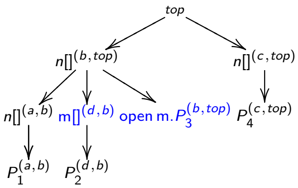
$$C = n[n[P_1] | m[P_2] | \text{open m}.P_3] \mid n[P_4]$$



(3)

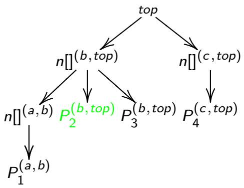
The basic dynamics

$C = n[n[P_1]|m[P_2]|\text{open m}.P_3] \mid n[P_4]$



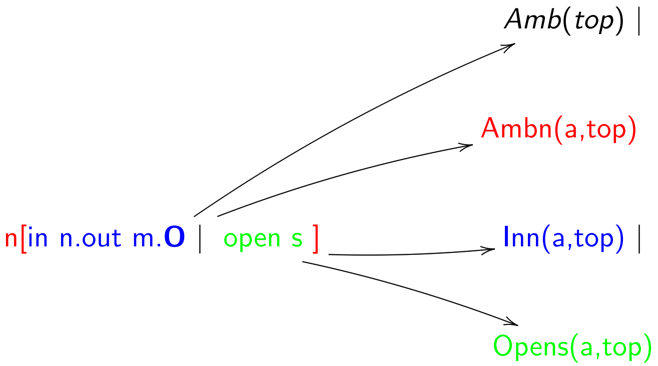
(3)

$D = n[n[P_1]|P_2|P_3] \mid n[P_4]$



(4)

The basic idea



Main encoding function

$$\mathcal{T} : \mathcal{P}_{MA} \times \mathcal{N} \rightarrow \mathcal{P}_{\pi}$$

$$\mathcal{T}(P, top) = (\nu\, \tilde{m})\mathcal{T}_a(P, top, top) | \mathrm{Amb(top)}$$

Main encoding function

$\mathcal{T} : \mathcal{P}_{MA} \times \mathcal{N} \to \mathcal{P}_\pi$

$$\mathcal{T}(P, top) = (\nu\, \tilde{m})\mathcal{T}_a(P, top, top) | \mathrm{Amb(top)}$$

where:

- $\tilde{m} = fn(\mathcal{T}_a(P, \mathrm{top}, \mathrm{top}))$,
- $\mathcal{T}_a : \mathcal{P}_{MA} \times \mathcal{N} \times \mathcal{N} \to \mathcal{P}_\pi$

An example

$$P = open\ n.\mathbf{0} | n[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

An example

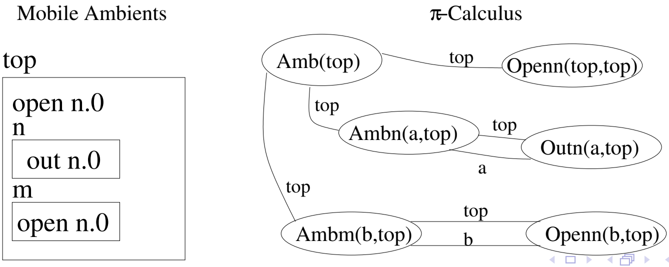
$$P = open\ n.\mathbf{0} | n[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

$$\begin{array}{rcl} \mathcal{T}(P, top) \equiv Q & = & (\nu\ top, a, b, \dots) \\ & & Amb(top) | Openn(top, top) \\ & | & Ambn(a, top) | Outn(a, top) \\ & | & Ambm(b, top) \mid Openn(b, top) \end{array}$$

An example

$$P = open\ n.\mathbf{0} | n[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

$$\begin{aligned}\mathcal{T}(P, top) \equiv Q \quad = \quad & (\nu\ top, a, b, \dots) \\ & Amb(top) | Openn(top, top) \\ | \quad & Ambn(a, top) | Outn(a, top) \\ | \quad & Ambm(b, top)\ |\ Openn(b, top)\end{aligned}$$



Encoding processes at work

$$P = \text{open } n.\mathbf{0} | n[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

Encoding processes at work

$$P = \text{open } n.\mathbf{0} | n[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

$$P \rightarrow P' \equiv out\ n.\mathbf{0} | m[open\ n.\mathbf{0}]$$

Encoding processes at work

$$P = \text{open n}.\mathbf{0}|\text{n}[out\ n.\mathbf{0}]|m[open\ n.\mathbf{0}]$$

$$P \rightarrow P' \equiv out\ n.\mathbf{0}|m[open\ n.\mathbf{0}]$$

Mimicking a capability:

Encoding processes at work

$$P = \text{open n}.\mathbf{0} | \text{n}[out\ n.\mathbf{0}] | m[open\ n.\mathbf{0}]$$

$$P \rightarrow P' \equiv out\ n.\mathbf{0} | m[open\ n.\mathbf{0}]$$

Mimicking a capability:

$$\begin{array}{rcl} \mathcal{T}(P, top) \stackrel{\tilde{\theta}}{\rightarrow}{}^{*} Q' & \equiv & Amb(top) \\ & | & \text{Opened(a,top,top)} | Outn(a, top) \\ & | & Ambm(m', top) | Openn(m', top) \end{array}$$

Acting as a link

The code:

$$
\begin{array}{rcl}
Opened(a, a', f') & = & a(x).\overline{x}\langle a', f'\rangle.Opened(a, a', f') \\
& & + \\
& & a(x, x').\overline{x}\langle a', f'\rangle.Opened(a, a', f')
\end{array}
$$

Running *encoding processes*

$$\mathcal{T}(P, top) = Q \xrightarrow{\theta} Q' \equiv (\nu top, a, \ldots)Amb(top)|\text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

Running *encoding processes*

$$\mathcal{T}(P, top) = Q \xrightarrow{\theta} Q' \equiv (\nu top, a, \ldots)Amb(top)|\text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

- Process Opened(a,top,top) has substituted the dissolved Ambn(a,top).
- The old coordinates will be updated as soon as Outn(a,top) will try to mimic the 'out n' capability.

Opened(a,top,top) only acts as a link between the old channel name **a** of the dissolved ambient, and the unique channel name **top** associated to the ambient which has taken its place.

Running *encoding processes*

$$\mathcal{T}(P, top) = Q \xrightarrow{\theta} Q' \equiv (\nu top, a, \ldots)Amb(top)|\text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

- Process Opened(a,top,top) has substituted the dissolved Ambn(a,top).
- The old coordinates will be updated as soon as Outn(a,top) will try to mimic the 'out n' capability.

Opened(a,top,top) only acts as a link between the old channel name **a** of the dissolved ambient, and the unique channel name **top** associated to the ambient which has taken its place.

$$\mathcal{T}(P', top) = Q'' \equiv \quad (\nu\ top, b, \ldots)\ Amb(top) \mid Outn(top, top) \mid Ambm(b, top)|Openn(b, top)$$

Encoding processes at work

Now we have that:

$$Q' \equiv Amb(top)|$$
$$\text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

Encoding processes at work

Now we have that:

$$Q' \equiv Amb(top)| \text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

while

Encoding processes at work

Now we have that:

$$Q' \equiv Amb(top)|$$
$$\text{Opened(a,top,top)}|Outn(a, top)|Ambm(m', top)|Openn(m', top)$$

while

$$\begin{aligned} \mathcal{T}(P', top) = Q'' \quad \equiv & \quad (\nu \; top, b) \; Amb(top) \\ | & \quad Outn(top, top) \\ | & \quad Ambm(b, top)|Openn(b, top) \end{aligned}$$

Q' differs from Q'' as it contains the subprocess $Opened(a, top, top)$ and different coordinates for $Outn(a, top)$.

Syntactic characterization

- The set of encoding π-calculus processes:

Syntactic characterization

- The set of encoding π-calculus processes:
  - $A\pi = \{Q \,|\, \exists P \in \mathcal{P}_{MA} \text{ s.t. } Q \equiv \mathcal{T}(P, top)\}$

Syntactic characterization

- The set of encoding π-calculus processes:
  - $A\pi = \{Q \,|\, \exists P \in \mathcal{P}_{MA} \text{ s.t. } Q \equiv \mathcal{T}(P, top)\}$
- The set of the derivatives of the encoding processes

Syntactic characterization

- The set of encoding π-calculus processes:
  - $A\pi = \{Q \,|\, \exists P \in \mathcal{P}_{MA} \text{ s.t. } Q \equiv \mathcal{T}(P, top)\}$
- The set of the derivatives of the encoding processes
  - $A\pi_{op} = \{Q' \,|\, Q \rightarrow^* Q',\, Q \in A\pi\ A\pi,\ Q' \equiv \Pi_{i=1}^{N}(\nu\tilde{m}_i)Ide_i(\tilde{m}_i)\}$

Syntactic characterization

- The set of encoding π-calculus processes:
  - $A\pi = \{Q \,|\, \exists P \in \mathcal{P}_{MA} \text{ s.t. } Q \equiv \mathcal{T}(P, top)\}$
- The set of the derivatives of the encoding processes
  - $A\pi_{op} = \{Q' \mid Q \rightarrow^{*} Q',\, Q \in A\pi\ A\pi,\ Q' \equiv \Pi_{i=1}^{N}(\nu \tilde{m}_i) Ide_i(\tilde{m}_i)\}$
  - i.e. those processes that differs from $A\pi$ because may contain *Opened*() subprocesses.

Syntactic characterization

The equivalence relation $\simeq$ deals with the presence of the identifiers *Opened*$(\dots)$ in the derivatives of translating processes.

Syntactic characterization

The equivalence relation $\simeq$ deals with the presence of the identifiers *Opened*$(\dots)$ in the derivatives of translating processes. It relates processes that contain *Opened*$()$ processes, with those processes that do not, but that may mimic the same capabilities:

If $R \in A_{\pi_{op}}$ (i.e. contains Opened() processes) and it exists a process $Q \in A_{\pi}$ (doesn't) and still may mimic the same set of capabilities

Syntactic characterization

The equivalence relation $\simeq$ deals with the presence of the identifiers *Opened*$(\ldots)$ in the derivatives of translating processes. It relates processes that contain *Opened*$()$ processes, with those processes that do not, but that may mimic the same capabilities:

If $R \in A_{\pi_{op}}$ (i.e. contains Opened() processes) and it exists a process $Q \in A_\pi$ (doesn't) and still may mimic the same set of capabilities then we write $R \simeq Q$.

Syntactic characterization

The equivalence relation $\simeq$ deals with the presence of the identifiers *Opened*$(\dots)$ in the derivatives of translating processes. It relates processes that contain *Opened*$()$ processes, with those processes that do not, but that may mimic the same capabilities:

If $R \in A_{\pi_{op}}$ (i.e. contains Opened() processes) and it exists a process $Q \in A_{\pi}$ (doesn't) and still may mimic the same set of capabilities then we write $R \simeq Q$.

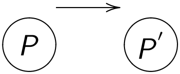
The notions of traces

A capability execution in MA: $P \;\rightarrow\; P'$
A capability mimicking in π-calculus:

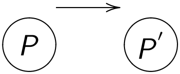
$$\mathcal{T}(P, top) \xrightarrow{\theta_1} \ldots \xrightarrow{\theta_n} Q'$$

- Q' records the effects of a capability simulation, $\xi = \theta_1 \ldots \theta_n$ is a *simulating trace*
- if $Q' \equiv \mathcal{T}(P, top)$ then we have an *aborting trace*

The simulation of a capability

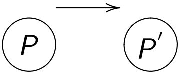


The simulation of a capability

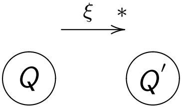


Simulating trace: ξ

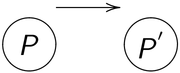
The simulation of a capability

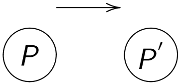


Simulating trace:ξ

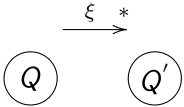


The simulation of a capability

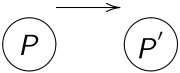




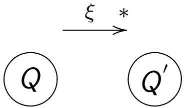
Simulating trace: ξ

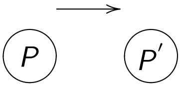


The simulation of a capability



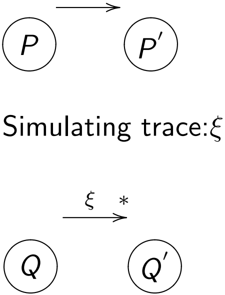
Simulating trace:ξ

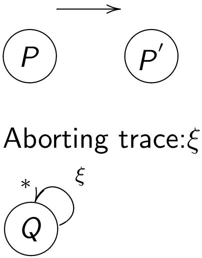




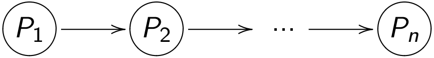
Aborting trace:ξ

The simulation of a capability

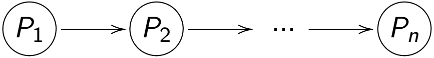




Ordered execution of traces

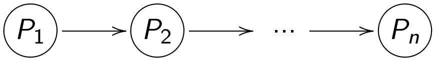


Ordered execution of traces

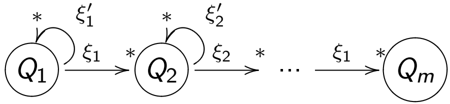


Assuming to execute orderly a trace per time.

Ordered execution of traces



Assuming to execute orderly a trace per time.



Weak soundness

Let P be a MA process,
if $\mathcal{T}(P, top) = Q \stackrel{\tilde{\theta}_a}{\longrightarrow}^* Q'$, then $\exists\ P'$, Q'' such that $Q' \stackrel{\tilde{\theta}_b}{\longrightarrow}^* Q''$, $Q'' \simeq \mathcal{T}(P', top)$ with $P \longrightarrow^* P'$.

Completeness

Let P be a MA process, if $P \longrightarrow^{*} P'$ then there exists a process Q' such that $\mathcal{T}(P, top) \stackrel{\tilde{\theta}}{\longrightarrow}{}^{*} Q'$ and $\mathcal{T}(P', top) \simeq Q'$.

Who I am - Where I am

A clear separation of the two information.

Who I am - Where I am

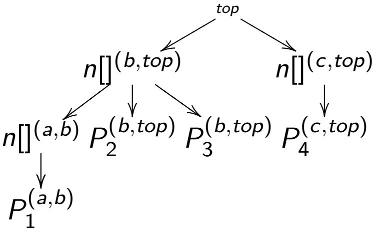
A clear separation of the two information.

$D = n[n[P_1]|P_2|P_3] \mid n[P_4]$

Who I am - Where I am

A clear separation of the two information.

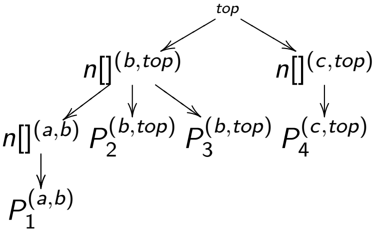
$D = n[n[P_1]|P_2|P_3] \mid n[P_4]$



Who I am - Where I am

A clear separation of the two information.

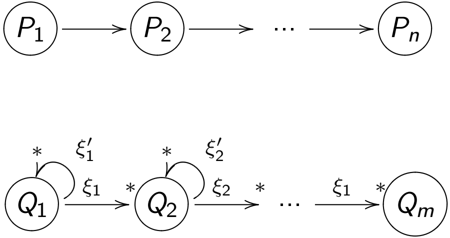
$D = n[n[P_1]|P_2|P_3] \mid n[P_4]$



The ambient construct delimits a computing environment, and also holds a precise position within a hierarchical structure, hence each ambient has a unique ‘address’.

The two transition systems

A clear relation between the two transition systems.



Behavioral characterization and LTL for MA

- A more elegant characterization between the two languages in terms of a behavioral equivalence, for example the testing equivalence.
- The understanding of the two roles of the ambients (Who-I-am,Where-I-am) could help in defining a simple LTL for MA.

Merci de votre attention