Separability, Expressiveness and Decidability in the Ambient Logic

AS mobilité - December 2002

Outline

1. From π to Mobile Ambients

2. Mobile Ambients Behaviour and Spatial Logics

3. Expressiveness of the Ambient Logic

4. Separability,Decidability

From the π-calculus to Mobile Ambients

A need for a new paradigm

- Scope extrusion expresses the evolving structure of network's topology...

- ...but is it realy enough for modelling notions like:
  ressources (servers, terminals, applets ...)
  network hierarchy (IP addresses, subnetworks, execution sites ...)
  realistic communication (packets, firewalls ...)

- to improve expressiveness, define another paradigm:
  Mobile Ambients

The Mobile Ambients paradigm [CarGor98]

- The basic notion is not names as in π anymore, but locations and sublocations (called <u>ambients</u>)

$$a[\ b[\,]|c[\,]\ ]\ |\ d[\,]$$

.

The Mobile Ambients paradigm [CarGor98]

- The basic notion is not names as in π anymore, but locations and sublocations (called ambients)

$$a[\ b[]|c[]\ ]\ |\ d[\,]$$

- The computation is not a name passing process anymore, but movement of locations

.
$$a[\mathsf{in}\, b]\,|\,b[\,]\ \longrightarrow\ b[a[\,]]$$

The Syntax

$$
\begin{array}{rcll}
\mathsf{cap} & \stackrel{def}{=} & \mathsf{in}\ n \mid \mathsf{out}\ n \mid \mathsf{open}\ n \mid (x) & \text{capabilities} \\
P & \stackrel{def}{=} & \mathbf{0} \mid n[P] \mid P_1|P_2 \mid !P \mid (\nu n)P & \text{spatial constructions} \\
 & & \mid \mathsf{cap}.P \mid \langle n \rangle & \text{temporal constructions}
\end{array}
$$

- spatial constructions : the process tree
- temporal constructions: evolution of trees

Semantics of the movement capabilities

In rule:

$$a[\mathsf{in}\ b.P_1|P_2]|b[P_3] \quad \longrightarrow \quad b[a[P_1|P_2]|P_3]$$

Out rule:

$$b[a[\mathsf{out}\ b.P_1|P_2]|P_3] \quad \longrightarrow \quad a[P_1|P_2] \mid b[P_3]$$

Open rule:

$$\mathsf{open}\ b.P_1|b[P_2] \quad \longrightarrow \quad P_1 \mid P_2$$

Semantics of communication

Comm rule:

$$(x)P \mid \langle n \rangle \quad \rightarrow \quad P\{n/x\}$$

Scope extrusions:

$$(\nu n)P \mid Q \quad \equiv \quad (\nu n)(P|Q) \quad (n \notin \mathsf{fn}(Q))$$

$$(\nu n)a[P] \quad \equiv \quad a[(\nu n)P] \quad (a \neq n)$$

Ambients Behaviour and Spatial Logic

Behaviour and Logic: the standard approach

- In the case of CCS or the π-calculus, we may define the semantics by means of a LTS

$$P \quad \xrightarrow{l} \quad Q$$

Behaviour and Logic: the standard approach

- In the case of CCS or the π-calculus, we may define the semantics by means of a LTS

$$P \xrightarrow{l} Q$$

- this allows one to define the behaviour of a process; bisimilarity relation:

.
$$P \approx Q$$

relates processes having the same behaviour.

Behaviour and Logic: the standard approach

- In the case of CCS or the π-calculus, we may define the semantics by means of a LTS

$$P \xrightarrow{l} Q$$

- this allows one to define the behaviour of a process; bisimilarity relation:

.

$$P \approx Q$$

relates processes having the same behaviour.

- Based on the LTS, we may introduce the Henessy-Milner logic with *action modalities* and *fixpoint recursion*:

.

$$\begin{array}{lll} P \models \langle \mathsf{a} \rangle.A & \text{iff} & \exists P'.\ P \xrightarrow{a} P' \ \wedge\ P' \models A \\ P \models \mu X.A & \text{iff} & P \models A\{\mu X.A\ /X\} \end{array}$$

Behaviour and Logic: the standard approach

- In the case of CCS or the π-calculus, we may define the semantics by means of a LTS

$$P \xrightarrow{l} Q$$

- this allows one to define the behaviour of a process; bisimilarity relation:

.
$$P \approx Q$$

relates processes having the same behaviour.

- Based on the LTS, we may introduce the Henessy-Milner logic with *action modalities* and *fixpoint recursion*:

.
$$\begin{array}{lll} P \models \langle \mathsf{a} \rangle.A & \text{iff} & \exists P'.\ P \xrightarrow{a} P' \ \wedge \ P' \models A \\ P \models \mu X.A & \text{iff} & P \models A\{\mu X.A\ /X\} \end{array}$$

- Behaviour and logic coincide: $=_L \ = \ \approx$

A behavioural semantics for Ambients?

- Some propositions of LTS have been introduced (Cardelli, Gordon, Henessy, Merro), but are not very natural. The problems are that reduction may operate at any nesting of ambients (and not at "top-level" like in π), and actions don't come with coactions (asynchrony).

A behavioural semantics for Ambients?

- Some propositions of LTS have been introduced (Cardelli, Gordon, Henessy, Merro), but are not very natural. The problems are that reduction may operate at any nesting of ambients (and not at "top-level" like in π), and actions don't come with coactions (asynchrony).

- Another notion of observational equivalence:

- A notion of barb: $P \Downarrow_n$ if $P \rightarrow^* n[P_1]|P_2$

A behavioural semantics for Ambients?

- Some propositions of LTS have been introduced (Cardelli, Gordon, Henessy, Merro), but are not very natural. The problems are that reduction may operate at any nesting of ambients (and not at "top-level" like in π), and actions don't come with coactions (asynchrony).

- Another notion of observational equivalence:

- A notion of barb: $P \Downarrow_n$ if $P \rightarrow^* n[P_1]|P_2$

-A barb congruence preorder: $P \sqsubseteq Q$ if for all C, n if $C\{P\} \Downarrow_n$, then $C\{Q\} \Downarrow_n$.

- $P \approx Q$ iff $P \sqsubseteq Q$ and $Q \sqsubseteq P$

How should we define behaviour for Ambients?

- Intersection types (Dezani,Coppo):

Types look like:

$$T \;::=\; T|T \;\Big|\; \mathsf{cap}.T \;\Big|\; \langle T^{-}\rangle.T \;\Big|\; (T^{-}).T \;\Big|\; a[T] \;\Big|\; T \wedge T \;\Big|\; \omega$$

- Description of the spatial behaviour using a spatial logic

The logical approach

- The behaviour is the evolution of *space structure*. The way HM-logic describes behaviour with action modalities, a logic for Ambients should describe behaviour by means of spatial connectives.

The logical approach

- The behaviour is the evolution of *space structure*. The way HM-logic describes behaviour with action modalities, a logic for Ambients should describe behaviour by means of spatial connectives.

- The Ambient Logic (AL) will reflect the spatial operators of the calculus:

ex: $a[\top] \mid b[c[0]]$

The logical approach

- The behaviour is the evolution of *space structure*. The way HM-logic describes behaviour with action modalities, a logic for Ambients should describe behaviour by means of spatial connectives.

- The Ambient Logic (AL) will reflect the spatial operators of the calculus:

ex: $a[\top] \mid b[c[0]]$

- AL includes classical logic:

ex: $\exists n.\ \ n[0] \mid (n[0]\ \vee\ \forall m.\neg m[0])$

The logical approach

- The behaviour is the evolution of *space structure*. The way HM-logic describes behaviour with action modalities, a logic for Ambients should describe behaviour by means of spatial connectives.

- The Ambient Logic (AL) will reflect the spatial operators of the calculus:

ex: $a[\top] \mid b[c[0]]$

- AL includes classical logic:

ex: $\exists n.\ \ n[0] \mid (n[0]\ \vee\ \forall m.\neg m[0])$

- AL should also express evolution of space structure: the $\Diamond$ modality

The logical approach

- The behaviour is the evolution of *space structure*. The way HM-logic describes behaviour with action modalities, a logic for Ambients should describe behaviour by means of spatial connectives.

- The Ambient Logic (AL) will reflect the spatial operators of the calculus:

ex: $a[\top] \mid b[c[0]]$

- AL includes classical logic:

ex: $\exists n.\ \ n[0] \mid (n[0] \ \vee \ \forall m. \neg m[0])$

- AL should also express evolution of space structure: the $\Diamond$ modality

- AL also has adjunct connectives:

- $. \triangleright .$ for $.|.$
- $.@n$ for $n[.]$

The satisfaction relation

Classical Logic

$P \models \mathcal{A} \wedge \mathcal{B},\ \neg\, \mathcal{A},\ \forall x.\mathcal{A},\ \top$ as usual

The satisfaction relation

Classical Logic

$P \models \mathcal{A} \wedge \mathcal{B},\ \neg\, \mathcal{A},\ \forall x.\mathcal{A},\ \top$ as usual

Intensional spatial connectives

$P \models \mathcal{A}_1 \mid \mathcal{A}_2$ iff $\exists\, P_1,\ P_2$ s.t. $P \equiv P_1|P_2$ and $P_i \models \mathcal{A}$

($\equiv$: *structural congruence*, almost syntactic equality)

The satisfaction relation

Classical Logic

$P \models \mathcal{A} \wedge \mathcal{B},\ \neg\ \mathcal{A},\ \forall x.\mathcal{A},\ \top$ as usual

Intensional spatial connectives

$P \models \mathcal{A}_1 \mid \mathcal{A}_2$ iff $\exists\ P_1,\ P_2$ s.t. $P \equiv P_1|P_2$ and $P_i \models \mathcal{A}$

($\equiv$: *structural congruence*, almost syntactic equality)

$P \models n[\mathcal{A}]$ iff $\exists\ P'$ s.t. $P \equiv n[P']$ and $P' \models \mathcal{A}$

$P \models \mathbf{0}$ iff $P \equiv \mathbf{0}$

The satisfaction relation

Classical Logic

$P \models \mathcal{A} \wedge \mathcal{B},\ \neg\, \mathcal{A},\ \forall x.\mathcal{A},\ \top$ as usual

Intensional spatial connectives

$P \models \mathcal{A}_1 \mid \mathcal{A}_2$ iff $\exists\ P_1,\ P_2$ s.t. $P \equiv P_1|P_2$ and $P_i \models \mathcal{A}$

($\equiv$: *structural congruence*, almost syntactic equality)

$P \models n[\mathcal{A}]$ iff $\exists\ P'$ s.t. $P \equiv n[P']$ and $P' \models \mathcal{A}$

$P \models \mathbf{0}$ iff $P \equiv \mathbf{0}$

Adjunct connectives

$P \models \mathcal{A} \triangleright \mathcal{B}$ iff $\forall\ Q$ s.t. $Q \models \mathcal{A}$, we have $P \mid Q \models \mathcal{B}$

$P \models \mathcal{A}\ @\ n$ iff $n[P] \models \mathcal{A}$

The satisfaction relation

Classical Logic

$P \models \mathcal{A} \wedge \mathcal{B},\ \neg\ \mathcal{A},\ \forall x.\mathcal{A},\ \top$ as usual

Intensional spatial connectives

$P \models \mathcal{A}_1 \mid \mathcal{A}_2$ iff $\exists\ P_1,\ P_2$ s.t. $P\ \equiv\ P_1|P_2$ and $P_i \models \mathcal{A}$

($\equiv$: *structural congruence*, almost syntactic equality)

$P \models n[\mathcal{A}]$ iff $\exists\ P'$ s.t. $P\ \equiv\ n[P']$ and $P' \models \mathcal{A}$

$P \models \mathbf{0}$ iff $P\ \equiv\ \mathbf{0}$

Adjunct connectives

$P \models \mathcal{A} \triangleright \mathcal{B}$ iff $\forall\ Q$ s.t. $Q \models \mathcal{A}$, we have $P \mid Q \models \mathcal{B}$

$P \models \mathcal{A}\ @\ n$ iff $n[P] \models \mathcal{A}$

Temporal connective

$P \models \Diamond\ \mathcal{A}$ iff $\exists\ P'$ s.t. $P \rightarrow^* P'$ and $P' \models \mathcal{A}$

Expressiveness of the Ambient Logic

What does the Ambient Logic speak about?

To which extent does AL talk about syntax?

This is not clear because:

- some elements of the syntax are present in the logic, but not all of them (capabilities, replication)
- evolution of processes: only the *"sometime"* modality ($\Diamond\mathcal{A}$)
- unusual adjunct connectives ($\mathcal{A}@n$, $\mathcal{A} \triangleright \mathcal{B}$)

Expressing capabilities

Formulas for possibility (intensional): [San01]

$$P \models \langle \mathsf{cap} \rangle.\mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

Expressing capabilities

Formulas for possibility (intensional): [San01]

$$P \models \langle \mathsf{cap} \rangle.\mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \ \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

Formulas for necessity (intensional):

$$((\mathsf{cap})).\mathcal{A} \ \stackrel{def}{=} \ \langle \mathsf{cap} \rangle.\mathcal{A} \ \wedge \ \neg \langle \mathsf{cap} \rangle.\neg \mathcal{A}$$

Using this, $P \models ((\mathsf{cap})).\mathcal{A}$ iff

$$\exists P_1, \quad P \ \equiv \mathsf{cap}.P_1, \text{ and whenever } P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2,\ P_2 \models \mathcal{A}$$

Expressing capabilities – an example

$$P \models \langle \mathsf{cap} \rangle . \mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

$$P \models ((\mathsf{cap})) . \mathcal{A} \quad \text{iff} \quad \exists P_1,\quad P \equiv \mathsf{cap}.P_1,\ \text{and whenever } P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2,\ P_2 \models \mathcal{A}$$

<u>ex:</u>

$$\langle \mathsf{open}\ n \rangle . \mathcal{A} \stackrel{def}{=} \mathsf{1Cap} \wedge \forall m. \left(n[m[0]] \triangleright \Diamond\ (\mathcal{A} | m[0]) \right)$$

Expressing capabilities – an example

$$P \models \langle \mathsf{cap} \rangle.\mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

$$P \models ((\mathsf{cap})).\mathcal{A} \quad \text{iff} \quad \exists P_1,\quad P \equiv \mathsf{cap}.P_1, \text{ and whenever } P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2,\ P_2 \models \mathcal{A}$$

<u>ex:</u>

$$\underbrace{\langle \mathsf{open}\ n \rangle.\mathcal{A}}_{P} \stackrel{def}{=} \mathsf{1Cap} \ \wedge\ \forall m.\ \big(\ n[m[0]] \triangleright \Diamond\ (\mathcal{A} | m[0])\ \big)$$

Expressing capabilities – an example

$$P \models \langle \mathsf{cap} \rangle.\mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

$$P \models ((\mathsf{cap})).\mathcal{A} \quad \text{iff} \quad \exists P_1,\quad P \equiv \mathsf{cap}.P_1, \text{ and whenever } P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2,\ P_2 \models \mathcal{A}$$

<u>ex:</u>

$$\langle \mathsf{open}\ n \rangle.\mathcal{A} \quad \stackrel{def}{=} \quad 1\mathsf{Cap} \quad \wedge \quad \forall m.\ \big(\ n[m[0]] \triangleright \Diamond\ (\mathcal{A}|m[0])\ \big)$$

$$P \mid n[m[0]]$$

Expressing capabilities – an example

$$P \models \langle \mathsf{cap} \rangle .\mathcal{A} \quad \text{iff} \quad \exists P_1, P_2.\ P \equiv \mathsf{cap}.P_1,\ P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2 \text{ and } P_2 \models \mathcal{A}$$

$$P \models ((\mathsf{cap})).\mathcal{A} \quad \text{iff} \quad \exists P_1,\quad P \equiv \mathsf{cap}.P_1, \text{ and whenever } P_1 \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} P_2,\ P_2 \models \mathcal{A}$$

<u>ex:</u>

$$\langle \mathsf{open}\ n \rangle .\mathcal{A} \stackrel{def}{=} \mathsf{1Cap} \wedge \forall m.\ \big(n[m[0]] \triangleright \Diamond\ (\mathcal{A} | m[0]) \big)$$

$$P \mid n[m[0]] \quad \rightarrow^{*} \quad P' \mid m[0] \quad \text{and } P' \models \mathcal{A}$$

Expressing replication

Given a formula $\mathcal{A}$ *"expressive enough"*, we may define a formula $!\mathcal{A}$ s.t. $P \models !\mathcal{A}$ iff

$$\begin{array}{rl} & \exists P_1, \ldots, P_r. \qquad P \;\equiv\; !P_1 |\ (!) P_2 | \ldots | (!) P_r \\ \text{and} & P_i \models \mathcal{A},\ i = 1 \ldots r \end{array}$$

N.B.: no infinitary construct available, instead we rely on $\Diamond$

Expressing replication

Given a formula $\mathcal{A}$ *"expressive enough"*, we may define a formula $!\mathcal{A}$ s.t. $P \models !\mathcal{A}$ iff

$$\begin{array}{rl} & \exists P_1, \ldots, P_r. \qquad P \equiv\ !P_1|\ (!)P_2|\ldots|(!)P_r \\ \text{and} & P_i \models \mathcal{A},\ i = 1 \ldots r \end{array}$$

N.B.: no infinitary construct available, instead we rely on $\Diamond$

The encoding (rather tedious):

$$!\mathcal{A} \quad \text{“}\stackrel{def}{=}\text{”} \quad \mathcal{A}^{\omega} \quad \wedge \quad \mathcal{A}^{pers}$$

$\mathcal{A}^{\omega}$: there are only copies of $\mathcal{A}$ at toplevel

$\mathcal{A}^{pers}$: there are infinitely many of them

Characteristic formulas

We may express all connectives of the calculus, so we may hope to be able to define *characteristic formulas*:

$$Q \models \mathfrak{F}_P \quad \text{iff} \quad Q =_L P$$

$Q =_L P$ iff P and Q satisfy the same formulas

We actually need an *image-finiteness* hypothesis:
$\longrightarrow$ subcalculus $\mathsf{MA}_{\mathsf{IF}}$: in any subterm cap.$P$, P is image-finite
Characteristic formulas can be defined on $\mathsf{MA}_{\mathsf{IF}}$

Separability, Decidability

A coinductive characterisation

$=_L$ coincides with *intensional bisimilarity,* $\simeq_{int}$:
whenever $P \simeq_{int} Q$,

$$
\begin{array}{lll}
P \equiv \mathbf{0} & \text{implies} & Q \equiv \mathbf{0} \\
P \equiv P_1 | P_2 & \text{implies} & Q \equiv Q_1 | Q_2 \ \text{ with } \ P_i \simeq_{int} Q_i \ \ (i = 1, 2) \\
P \equiv n[P_1] & \text{implies} & Q \equiv n[Q_1] \\
P \stackrel{\mathsf{cap}}{\rightarrow} P_1 & \text{implies} & Q \stackrel{\mathsf{cap}}{\rightarrow} \stackrel{\langle \mathsf{cap} \rangle}{\Longrightarrow} Q_1 \text{ with } P_1 \simeq_{int} Q_1
\end{array}
$$

A coinductive characterisation

$=_L$ coincides with *intensional bisimilarity,* $\simeq_{int}$:
whenever $P \simeq_{int} Q$,

$$
\begin{array}{lllllll}
P & \equiv & \mathbf{0} & \text{implies} & Q & \equiv & \mathbf{0} \\
P & \equiv & P_1|P_2 & \text{implies} & Q & \equiv & Q_1|Q_2 \ \text{ with } \ P_i \simeq_{int} Q_i \ \ (i = 1,2) \\
P & \equiv & n[P_1] & \text{implies} & Q & \equiv & n[Q_1] \\
P & \stackrel{\mathsf{cap}}{\rightarrow} & P_1 & \text{implies} & Q & \stackrel{\mathsf{cap}}{\rightarrow} & \stackrel{\langle\mathsf{cap}\rangle}{\Longrightarrow} Q_1 \text{ with } P_1 \simeq_{int} Q_1
\end{array}
$$

- correction ($\simeq_{int} \subseteq =_L$): follows from congruence

A coinductive characterisation

$=_L$ coincides with *intensional bisimilarity,* $\simeq_{int}$:
whenever $P \simeq_{int} Q$,

$$\begin{array}{lllll} P \equiv \mathbf{0} & \text{implies} & Q \equiv \mathbf{0} & & \\ P \equiv P_1|P_2 & \text{implies} & Q \equiv Q_1|Q_2 & \text{with} & P_i \simeq_{int} Q_i \ (i=1,2) \\ P \equiv n[P_1] & \text{implies} & Q \equiv n[Q_1] & & \\ P \stackrel{\mathsf{cap}}{\rightarrow} P_1 & \text{implies} & Q \stackrel{\mathsf{cap}}{\rightarrow} \stackrel{\langle\mathsf{cap}\rangle}{\Longrightarrow} Q_1 & \text{with} & P_1 \simeq_{int} Q_1 \end{array}$$

- correction ($\simeq_{int} \subseteq =_L$): follows from congruence
- completeness ($=_L \subseteq \simeq_{int}$):

holds without image-finiteness hypothesis (on full MA)

Stuttering – *imprecise capabilities*

When $P \models \langle \text{in } n \rangle.\mathcal{A}$, there exist P', P'' s.t. $P \equiv \text{in } n.P'$ and

$$P' \xrightarrow{(\text{out } n, \text{in } n)^*} P'' \qquad \textit{(stuttering)} \qquad \text{and } P'' \models \mathcal{A}$$

Stuttering – *imprecise capabilities*

When $P \models \langle \text{in } n \rangle.\mathcal{A}$, there exist P', P'' s.t. $P \equiv \text{in } n.P'$ and

$$P' \xrightarrow{(\text{out } n, \text{in } n)^*} P'' \qquad \textit{(stuttering)} \qquad \text{and } P'' \models \mathcal{A}$$

<u>Consequence:</u>

$$P_1 \xrightarrow{(\text{out } n, \text{in } n)^*} P_2 \xrightarrow{(\text{out } n, \text{in } n)^*} P_1 \quad \text{iff} \quad \text{in } n.P_1 =_L \text{in } n.P_2$$

Stuttering – *imprecise capabilities*

When $P \models \langle \mathsf{in}\ n \rangle.\mathcal{A}$, there exist P', P'' s.t. $P \equiv \mathsf{in}\ n.P'$ and

$$P' \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P'' \qquad \textit{(stuttering)} \qquad \text{and } P'' \models \mathcal{A}$$

<u>Consequence:</u>

$$P_1 \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P_2 \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P_1 \quad \text{iff} \quad \mathsf{in}\ n.P_1 =_L \mathsf{in}\ n.P_2$$

Another subcalculus, $\mathsf{MA}^{\mathsf{syn}}_{\mathsf{IF}}$: in any subterm cap.$P$, P is finite

Stuttering – *imprecise capabilities*

When $P \models \langle \mathsf{in}\ n \rangle.\mathcal{A}$, there exist P', P'' s.t. $P \equiv \mathsf{in}\ n.P'$ and

$$P' \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P'' \qquad \textit{(stuttering)} \qquad \text{and } P'' \models \mathcal{A}$$

<u>Consequence:</u>

$$P_1 \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P_2 \xrightarrow{(\mathsf{out}\ n, \mathsf{in}\ n)^*} P_1 \quad \text{iff} \quad \mathsf{in}\ n.P_1 =_L \mathsf{in}\ n.P_2$$

Another subcalculus, $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$: in any subterm cap.$P$, P is finite

- $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}} \subset \mathsf{MA}_{\mathsf{IF}}$ (finite, hence image-finite)
- On $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$, $\quad \mathsf{in}\ n.P_1 =_L \mathsf{in}\ n.P_2 \quad$ iff $\quad P_1 =_L P_2$

The spectrum of separation of AL

- on $\mathsf{MA}^{\mathsf{syn}}_{\mathsf{IF}}$, $=_L$ $=$ $\equiv$

The spectrum of separation of AL

- on $\mathrm{MA}_{\mathrm{IF}}^{\mathrm{syn}}$, $=_L \quad = \quad \equiv$
- this does not hold on $\mathrm{MA}_{\mathrm{IF}}$

The spectrum of separation of AL

- on $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$, $\quad =_L \quad = \quad \equiv$
- this does not hold on $\mathsf{MA}_{\mathsf{IF}}$

$$
\begin{aligned}
P_0 &\stackrel{def}{=} \; !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid n[\mathbf{0}] \\
P_1 &\stackrel{def}{=} \; !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid \mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}]
\end{aligned}
$$

then $\quad \mathsf{out}\ n.P_0 =_L \mathsf{out}\ n.P_1$

The spectrum of separation of AL

- on $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$, $=_L \quad = \quad \equiv$
- this does not hold on $\mathsf{MA}_{\mathsf{IF}}$

$$P_0 \stackrel{def}{=} !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid n[\mathbf{0}]$$
$$P_1 \stackrel{def}{=} !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid \mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}]$$

then $\mathsf{out}\ n.P_0 =_L \mathsf{out}\ n.P_1$

- without image-finiteness, $=_L$ is undecidable

The spectrum of separation of AL

- on $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$, $\quad =_L \;=\; \equiv$

- this does not hold on $\mathsf{MA}_{\mathsf{IF}}$

$$P_0 \stackrel{def}{=} !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid n[\mathbf{0}]$$
$$P_1 \stackrel{def}{=} !\mathsf{open}\ n.\mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}] \mid \mathsf{in}\ n.\mathsf{out}\ n.n[\mathbf{0}]$$

then $\quad \mathsf{out}\ n.P_0 =_L \mathsf{out}\ n.P_1$

- without image-finiteness, $=_L$ is undecidable

proof: we define $P_1, P_2 \in \mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$ such that $P_1 \rightarrow P_2$,

but $P_2 \rightarrow^* P_1$ is undecidable.

Then $\mathsf{open}\ n.P_1 \stackrel{?}{=}_L \mathsf{open}\ n.P_2$ is undecidable.

Completeness: key ideas

We may capture the first layer of capabilities in a process (*active context*):
in $n.a[b[\mathbf{0}]] \mid !b[\text{open } n.\text{out } n] \quad \rightsquigarrow \quad \text{in } n.[]_1 \mid !b[\text{open } n.[]_2]$

the rest of the term (*continuations*) is preserved under reduction:

$$P \to Q \quad \Rightarrow \quad \mathsf{cont}(Q) \subseteq \mathsf{cont}(P)$$

Lemma (Partial characteristic formulas)
For all P, Q, there is $F_{P,Q}$ such that $P \models F_{P,Q}$ and for all Q' such that $Q \to^* Q'$,

$$Q' \models F_{P,Q} \quad \text{iff} \quad Q' \simeq_{int} P$$

Theorem (Completeness) $\quad =_L \quad \subseteq \quad \simeq_{int}.$

Conclusion: Separability of AL

- AL expresses more than behaviour ($=_L \subsetneq \approx$); for *most of* processes,

$$P =_L Q \qquad \text{iff} \qquad P \equiv Q$$

- However, for some *extreme* processes, the result fails because the $\Diamond$ has a weak semantic ($\longrightarrow^*$ instead of $\longrightarrow$).

- The imprecisions due to the many-steps semantics:

- η-convertibility: $(x)\big(\langle x\rangle|(y)P\big) \;=_L\; (y)P$
- stuttering: in $n.P =_L$ in $n.Q$ iff $P \xrightarrow{(\text{out } n, \text{in } n)^*} Q \xrightarrow{(\text{out } n, \text{in } n)^*} P$

Decidability issues in AL

- Model-checking and validity are mutually dependent ($\triangleright$,F_P)

- In the general case, both are undecidable (Talbot,Charatonik)

A short proof: $P \models F_Q \wedge \Diamond F_R$ and $\vdash F_Q \rightarrow \Diamond F_R$ boils down to decide wether $Q \rightarrow^* R$.

- Some cases where decidability has been obtained:

- finite control Ambients, logic without $\triangleright$ [ChaGorTal02]
- static trees, logic without $\forall$ and $\Diamond$ [CalCarGor02]

- Logical equivalence ($=_L$) is not decidable

in the general case, because of stuttering [HirLozSan02], while still being "very close" to $\equiv$ which is decidable (DalZilio)

Extensions

Adding communication

$$\langle n \rangle \mid (x).P \quad \longrightarrow \quad P_{\{x := n\}}$$

Adding communication

$$\langle n \rangle \mid (x).P \quad \longrightarrow \quad P_{\{x:=n\}}$$

- messages and receptions may be captured using formulas

Adding communication

$$\langle n \rangle \mid (x).P \quad \longrightarrow \quad P_{\{x:=n\}}$$

- messages and receptions may be captured using formulas
- as before:

▷ image-finiteness $\Rightarrow$ characteristic formulas

▷ completeness: no need of image-finiteness

Adding communication

$$\langle n \rangle \mid (x).P \quad \longrightarrow \quad P_{\{x:=n\}}$$

- messages and receptions may be captured using formulas
- as before:
- ▷ image-finiteness $\Rightarrow$ characteristic formulas
- ▷ completeness: no need of image-finiteness
- $\mathsf{MA}^{\mathsf{syn}}_{\mathsf{IF}}$: $=_L$ coincides with $\equiv_\eta$, i.e. $\equiv$ on η-normal terms

$$(x).\Big(\langle x \rangle | (y).P\Big) \rightarrow_\eta (y).P$$

Adding name restriction

- this extension is less clear
- new logical connectives $n \circledR \mathcal{A}$ and $\text{И}n.\mathcal{A}$ [CarGor01]
- we believe that:

▷ logical equivalence is still $\equiv$ on $\mathsf{MA}_{\mathsf{IF}}^{\mathsf{syn}}$ for η-normalized terms

▷ characteristic formulas exist

▷ completeness only holds under image-finiteness

Conclusion

Main contributions

- evidence of the strong expressiveness of AL
- adjuncts are important in this setting
- characterisations of $=_L$ (coinductive and inductive)
- connections with other works about decidability in AL

$\longrightarrow$ to what extend do our technical developments (encoding of persistence, completeness proof) depend on the specific calculus of Mobile Ambients?

Current investigations

- Decidability with $\triangleright$: what is tractable?

- The π-calculus logic: what about encoding capabilities? (We have results)

- Less intensionnal logics: is there a way to define a "more behavioural" $=_L$?

Annex

Capability formulas

1Comp	$\stackrel{def}{=}$	$\neg 0 \quad \wedge \quad 0 \Vert 0$	
1Cap	$\stackrel{def}{=}$	$1\text{Comp} \quad \wedge \quad \neg\exists x.\ x[\top]$	
$\langle \text{in } n \rangle.\mathcal{A}$	$\stackrel{def}{=}$	$1\text{Cap} \quad \wedge \quad \forall x.\ \big(n[0] \triangleright \Diamond\ n[x[\mathcal{A}]]\big)@x$	
$\langle \text{out } n \rangle.\mathcal{A}$	$\stackrel{def}{=}$	$1\text{Cap} \quad \wedge \quad \forall m.\ \big((\Diamond m[\mathcal{A}] \vert n[0])@n\big)@m$	
$\langle \text{open } n \rangle.\mathcal{A}$	$\stackrel{def}{=}$	$1\text{Cap} \quad \wedge \quad \forall m.\ \big(\ n[m[0]] \triangleright \Diamond\ m[0] \vert \mathcal{A}\ \big)$	
$((\text{cap})).\mathcal{A}$	$\stackrel{def}{=}$	$\langle \text{cap} \rangle.\top \ \wedge\ \neg\langle \text{cap} \rangle.\neg\mathcal{A}$	for any capability cap

Characteristic formulas

$\mathcal{F}_{\mathbf{0}}$	$\stackrel{def}{=}$	0	$\mathcal{F}_{P\mid Q}$	$\stackrel{def}{=}$	$\mathcal{F}_P \mid \mathcal{F}_Q$
$\mathcal{F}_{n[P]}$	$\stackrel{def}{=}$	$n[\mathcal{F}_P]$	$\mathcal{F}_{\mathsf{cap}.P}$	$\stackrel{def}{=}$	$\langle \mathsf{cap} \rangle . \mathcal{F}_P \ \wedge \ ((\mathsf{cap})). \bigvee_{\{P',\ P \longrightarrow^{*} P'\}_{/\simeq_{int}}} \mathcal{F}_{P'}$
$\mathcal{F}_{!n[P]}$	$\stackrel{def}{=}$	$\mathsf{Rep}_{n[]}(\mathcal{F}_P)$	$\mathcal{F}_{!\mathsf{cap}.P}$	$\stackrel{def}{=}$	$\mathsf{Rep}_{\mathsf{cap}}(\mathcal{F}_{\mathsf{cap}.P})$