Making Ambients More Robust

Xudong Guan, Yiling Yang, Jinyuan You

Dept. of Computer Sci. & Eng., Shanghai Jiaotong Univ., China, 200030

{guan-xd, yang-yl, you-jy}@cs.sjtu.edu.cn

Abstract

Mobile Safe Ambients (SA) was proposed in order to remove the grave interference in the Mobile Ambient calculus. But the coactions introduced in SA also bring some security breaches. In this paper, a similar calculus called Robust Ambients (ROAM) was proposed as a more rational substitute for SA. Through specifying the parameters of the coactions, the ambients in ROAM are more robust against malicious tampering. The encoding of polyadic asynchronous π-calculus in ROAM shows that ROAM does not loss the strong expressiveness of its ancestors. The type system for ROAM proposed here also shows that the new calculus has some very good properties.

1. Introduction

The Mobile Ambient calculus (MA) [1] is proposed as a good way of expressing both communication and location concepts in a simple form. However, as Levi and Sangiorgi have pointed out in [4], there exists a kind of dangerous interference in the original ambient calculus. In their paper, a new calculus call Mobile Safe Ambients (SA) with its type system was presented in order to eliminate this kind of interference. But the coactions introduced in SA are very vulnerable to tampering in that they could be easily consumed by some malicious third parties. For example:

$n[in\ m.\overline{open}\ n.P] \mid m[\overline{in}\ m.open\ n.Q] \mid h[in\ m]$ (1)

$m[\overline{out}\ m.in\ n.P \mid n[out\ m.\overline{in}\ n.Q] \mid h[out\ m]]$ (2)

In (1), *n* is going to enter *m* and be opened there, while *m* is also willing to accept *n* and open it. But *m*'s sibling ambient *h* can easily destroy this protocol by maliciously using up the "$\overline{in}\ m$" capability to forbid the legal ambient *n* to enter. In (2) the same kind of tampering will happen. These are all because that there is no restriction on who should participate in the reduction and consume the coaction.

This paper alters the syntax of SA to explicitly name the ambient who could participate in the reduction. And yield a more robust ambient calculus with no grave interference – the Robust Ambient (ROAM). Expressiveness is also an important factor in evaluating a calculus. SA has strong expressiveness in that it could code MA by saturating every ambient in SA with unlimited availability of coactions. Later we will demonstrate by encoding the polyadic asynchronous π-calculus that restricting the parameter of the coactions will not reduce the expressive power of ROAM.

The rest of the paper is organized as follows: Section 2 gives an informal analysis of the outcome of the syntax of ROAM. Section 3 presents the syntax and reduction rules of ROAM. Section 4 gives a type system for ROAM. After giving the encoding of the firewall-crossing example and the polyadic asynchronous π-calculus in ROAM, the paper concludes and gives the future work.

2. Informal analysis

Coactions are the key to prevent the grave interference in MA. But the parameter of each coaction should be set properly. In SA, the parameter of each coaction is simply set to be the surrounding ambient name. This guarantees that MA will be easily encoded in SA by saturating every ambient n with unlimited number of coactions ($!\overline{in}\ n \mid !\overline{out}\ n \mid !\overline{open}\ n$). But, as we show in section 1, that this is the cause of the third party tampering. To gain more robustness, the parameter of each coaction should be further considered.

i). $\overline{in}$: If there are more that one external ambients waiting to enter, there should be something designating the one allowed. So we should utilize the parameter of $\overline{in}$: $\overline{in}\ n$ means that only the external ambient named *n* could enter.

ii). $\overline{out}$: Comparatively, we should designate which ambient to go out when there are more that one ambients competing for one $\overline{out}$ capability. So

$\overline{out}\ n$ means that only the child ambient named n could leave.

iii). $\overline{open}$: On the contrary, the parent ambient is the only one to use the $\overline{open}$ capability, so there is no need the place an parameter after it.

In addition, for the normal capability *out*, the only place to go is to exit the parent ambient, so the parameter behind it is also superfluous and could be omitted.

With the above analysis, we could reconstruct the above example (1) and (2) in the new calculus as follows:

$n[in\ m.\overline{open}.P] \mid m[\overline{in}\ n.open\ n.Q] \mid h[in\ m]$ (3)

$m[\overline{out}\ n.in\ n.P \mid n[out.\overline{in}\ m.Q] \mid h[out]]$ (4)

In this condition, the ambient h could not maliciously consume the coactions $\overline{in}\ n$ and $\overline{out}\ n$ any more.

With the above informal analysis, we now formalize ROAM in the next section.

3. The ROAM calculus

3.1. Syntax

Let ***N*** be a set of names ranged over by n. ROAM defines the processes set ***Proc*** ranged over by P and capabilities set ***Cap*** ranged over by M as follows:

P ::= $\mathbf{0} \mid (\nu n{:}W)P \mid P \mid Q \mid !P \mid M[P] \mid M.P \mid (n_1{:}W_1, \ldots, n_k{:}W_k).P \mid \langle M_1, \ldots, M_k\rangle$

M::= $\varepsilon \mid n \mid \mathbf{in}\ M \mid \overline{\mathbf{in}}\ M \mid \mathbf{out} \mid \overline{\mathbf{out}}\ M \mid \mathbf{open}\ M \mid \overline{\mathbf{open}} \mid M.M'$

To avoid too many parentheses, we define "|" to have the least precedence. As conventional, in process $(\nu n{:}W)P$ and $(n_1{:}W, \ldots, n_k{:}W).Q$, we say that the name n is occurred bounded in P and names n_1, …, and n_k are occurred bounded in Q. Otherwise, any occurrence of the name is free. Function $fn(P)$ is used to return a set containing all the free names occurred in process P. Also, the well known α-convention is used implicitly as need to avoid name capturing.

3.2. Substitution

We define substitution $P\{M/n\}$, $M'\{M/n\}$ over processes and capabilities respectively as follows:

(Sub-Eps) $\varepsilon\{M'/n\} = \varepsilon$
(Sub-n) $n'\{M'/n\} = n', \quad n' \neq n$
$n'\{M'/n\} = M', \quad n' = n$
(Sub-*in*) $(in\ M)\{M'/n\} = in\ (M\{M'/n\})$
(Sub-$\overline{in}$) $(\overline{in}\ M)\{M'/n\} = \overline{in}\ (M\{M'/n\})$
(Sub-*out*) $out\ \{M'/n\} = out$
(Sub-$\overline{out}$) $(\overline{out}\ M)\{M'/n\} = \overline{out}\ (M\{M'/n\})$
(Sub-*open*) $(open\ M)\{M'/n\} = open\ (M\{M'/n\})$
(Sub-$\overline{open}$) $\overline{open}\ \{M'/n\} = \overline{open}$
(Sub-Path) $(M_1.M_2)\{M'/n\}=(M_1\{M'/n\}).(M_2\{M'/n\})$
(Sub-Inact) $\mathbf{0}\ \{M'/n\} = \mathbf{0}$
(Sub-Res) $((\nu n'{:}W)P)\{M'/n\} = (\nu n'{:}W)(P\{M'/n\})$, $n' \neq n$ *and* $n' \notin fn(M')$
$((\nu n'{:}W)P)\{M'/n\} = (\nu n'{:}W)P$, $n' = n$
(Sub-Par) $(P|Q)\{M'/n\} = (P\{M'/n\})|(Q\{M'/n\})$
(Sub-Rep) $(!P)\{M'/n\} = !(P\{M'/n\})$
(Sub-Amb) $(M[P])\{M'/n\} = (M\{M'/n\})[P\{M'/n\}]$
(Sub-Act) $(M.P)\{M'/n\} = (M\{M'/n\}).(P\{M'/n\})$
(Sub-Abs) $((n_1{:}W_1, \ldots, n_k{:}W_k).P)\{M'/n\}$
$= (n_1{:}W_1, \ldots, n_k{:}W_k).(P\{M'/n\})$
, $n' \notin \{n_1, \ldots, n_k\}$ *and* $n' \notin fn(M')$
$((n_1{:}W_1, \ldots, n_k{:}W_k).P)\{M'/n\}$
$= (n_1{:}W_1, \ldots, n_k{:}W_k).P$, $n' \in \{n_1, \ldots, n_k\}$
(Sub-Msg) $(\langle M_1, \ldots, M_k\rangle)\{M'/n\}$
$= \langle M_1\{M'/n\}, \ldots, M_k\{M'/n\}\rangle$

For the purpose of simplification, in the rest of the paper, we use $P\{M_1/n_1, \ldots, M_k/n_k\}$ to denote $(\ldots(P\{M_1/n_1\})\ \ldots)\{M_k/n_k\}$ and $M'\{M_1/n_1, \ldots, M_k/n_k\}$ to denote $(\ldots(M'\{M_1/n_1\})\ \ldots)\{M_k/n_k\}$.

3.3. Structural Congruence

To denote structural congruence, the relation "≡"over set ***Proc*** is defined as follows:

(Struct Ref) $P \equiv P$
(Struct Symm) $P \equiv Q \Rightarrow Q \equiv P$
(Struct Trans) $P \equiv Q, Q \equiv R \Rightarrow P \equiv R$
(Struct Res) $P \equiv Q \Rightarrow (\nu n{:}W)P \equiv (\nu n{:}W)Q$
(Struct Par) $P \equiv Q \Rightarrow P \mid R \equiv Q \mid R$
(Struct Rep) $P \equiv Q \Rightarrow !P \equiv !Q$
(Struct Amb) $P \equiv Q \Rightarrow M[P] \equiv M[Q]$
(Struct Act) $P \equiv Q \Rightarrow M.P \equiv M.Q$
(Struct Abs) $P \equiv Q \Rightarrow (n_1{:}W_1, \ldots, n_k{:}W_k).P \equiv (n_1{:}W_1, \ldots, n_k{:}W_k).Q$
(Struct Par Comm) $P \mid Q \equiv Q \mid P$
(Struct Par Assoc) $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$
(Struct Rep) $!P \equiv P \mid !P$
(Struct Res Res) $(\nu n{:}W)\ (\nu m{:}V)P \equiv (\nu m{:}V)\ (\nu n{:}W)P$, if $n \neq m$
(Struct Res Par) $(\nu n{:}W)\ (P \mid Q) \equiv P \mid (\nu n{:}W)Q$, if $n \notin fn(P)$
(Struct Res Amb) $(\nu n{:}W)m[P] \equiv m[(\nu n{:}W)P]$, if $n \neq m$
(Struct Zero Par) $P \mid \mathbf{0} \equiv P$
(Struct Zero Res) $(\nu n{:}W)\mathbf{0} \equiv \mathbf{0}$
(Struct Zero Rep) $!\mathbf{0} \equiv \mathbf{0}$
(Struct Eps) $\varepsilon.P \equiv P$
(Struct Path) $(M.M').P \equiv M.M'.P$

3.4. Reduction

ROAM has the following reduction rules.

(R-*in*) $n[\overline{in}\ m.P_1|P_2] \mid m[in\ n.Q_1|Q_2] \rightarrow n[P_1|P_2|\ m[Q_1|Q_2]]$

(R-*out*) $n[\overline{out}\ m.P_1|P_2|\ m[out.Q_1|Q_2]] \rightarrow n[P_1|P_2] \mid m[Q_1|Q_2]$

(R-*open*) $open\ n.P \mid n[\overline{open}\ .Q_1 \mid Q_2] \rightarrow P \mid Q_1 \mid Q_2$

(R-Msg) $\langle M_1, \ldots, M_k\rangle \mid (n_1{:}W_1, \ldots, n_k{:}W_k).P \rightarrow P\{M_1/n_1, \ldots, M_k/n_k\}$

(R-Res) $$\frac{P \rightarrow P'}{(\nu n:W)P \rightarrow (\nu n:W)P'}$$

(R-Par) $$\frac{P \rightarrow P'}{P \mid Q \rightarrow P' \mid Q}$$

(R-Amb) $$\frac{P \rightarrow P'}{n[P] \rightarrow n[P']}$$

(R-≡) $$\frac{P \equiv P'\ \ P' \rightarrow P''\ \ P'' \equiv P'''}{P \rightarrow P'''}$$

The major changes to the reduction rules is that the reductions are controlled more strictly by both parties involved. The action grantor is able to specify which ambient is allowed to perform the action granted.

4. Types for ROAM

Our type system is benefited directly from [4], in which the notion of mobility and single-threadness for processes were introduced. In our type system, each capability and process has two attributes: the mobility and the number of threads. They are denoted by α and β respectively in the type expression. The *ambient* and *capability* types for capabilities and the message exchanged type for processes are just the same as what were introduced in the original type system for ambient calculus [2, 3].

4.1. Type Grammar

First we define set ***Alpha*** to be {Ω, ↑} and let α to range over it; we define set ***Beta*** to be {0, 1, 1^+, n} and let β to range over it. Then the type grammar of ROAM is defined as follows:

W	**::=** $V\ \alpha\ \beta$	, type for capabilities
V	**::=**	two kinds of capabilities:
	Amb[*S*]	, 1. Ambient, e.g. n in $n[P]$
	Cap[*S*]	, 2. Capability, e.g. *open n*, *in n*, etc.
T	**::=** $S\ \alpha\ \beta$	, type for processes
S	**::=**	two kinds of message exchanges:
	$W_1 \times \ldots \times W_k$	, 1. exchanges tuple (W_1, …, W_k)
	Shh	, 2. exchanges nothing (when k = 0)
α	**::=**	two kinds of mobility attribute
	↑	, 1. mobile (contains: *in* or *out*)
	Ω	, 2. immobile (otherwise)
β	**::=**	number of threads
	n	, 1. greater than 1
	1^+	, 2. equals 1, but end with *open n*
	1	, 3. equals 1, not end with *open n*
	0	, 4. equals 0

We define four operators on ***Alpha*** and ***Beta*** in the following tables:

i). binary operator ⊗

$\alpha_1 \otimes \alpha_2$:

α_1 \ α_2	↑	Ω
↑	↑	↑
Ω	↑	Ω

ii). binary operator ∘

$\beta_1 \circ \beta_2$:

β_1 \ β_2	n	1^+	1	0
n	n	n	n	n
1^+	n	n	n	1^+
1	n	1^+	1	1
0	n	1^+	1	0

iii). binary operator ⊕

$\beta_1 \oplus \beta_2$:

β_1 \ β_2	n	1^+	1	0
n	n	n	n	n
1^+	n	n	n	1^+
1	n	n	n	1
0	n	1^+	1	0

iv). unitary operator ↑

β	$\beta^{\uparrow}$
n	n
1^+	1^+
1	1^+
0	1

Also we define some relations :

i). We define relation "<" on set ***Alpha*** to be { Ω<↑ } and "≤" the reflective and transitive closure of "<". Obviously, we have the following properties of "≤":

Property 3-1(Symmetry): $\alpha_1 \otimes \alpha_2 = \alpha_2 \otimes \alpha_1$

Property 3-2(Associativeness): $(\alpha_1 \otimes \alpha_2) \otimes \alpha_3 = \alpha_1 \otimes (\alpha_2 \otimes \alpha_3)$

Property 3-3: $\alpha_1 \otimes \alpha_1 = \alpha_1$

Property 3-4: $\alpha_1 \otimes \alpha_2 = \alpha_3 \Rightarrow \alpha_1 \le \alpha_3 \wedge \alpha_2 \le \alpha_3$

Property 3-5: $\alpha_1 \le \alpha_2 \Rightarrow \alpha_1 \otimes \alpha_3 \le \alpha_2 \otimes \alpha_3$

ii). We define relation "<" on set ***Beta*** to be {0 < 1, 1<1^+, 1^+ <n} and "≤" the union of the reflective and transitive closure of "<" and {1^+≤1}. We also have some properties of "≤" as well:

Property 3-6(Associativeness): $(\beta_1 \circ \beta_2) \circ \beta_3 = \beta_1 \circ (\beta_2 \circ \beta_3)$

Property 3-7: $\beta_1 \circ \beta_2 = \beta_3 \Rightarrow \beta_1 \le \beta_3 \wedge \beta_2 \le \beta_3$

Property 3-8: $\beta_1 \le \beta_2 \Rightarrow \beta_1 \circ \beta_3 \le \beta_2 \circ \beta_3$

Property 3-9(Symmetry): $\beta_1 \oplus \beta_2 = \beta_2 \oplus \beta_1$

Property 3-10(Associativeness): $(\beta_1 \oplus \beta_2) \oplus \beta_3 = \beta_1 \oplus (\beta_2 \oplus \beta_3)$

Property 3-11: $\beta_1 \oplus \beta_1 = \beta_1 \oplus (\beta_1 \oplus \beta_1)$

Property 3-12: $\beta_1 \oplus \beta_2 = \beta_3 \Rightarrow \beta_1 \le \beta_3 \wedge \beta_2 \le \beta_3$

Property 3-13: $\beta_1 \le \beta_2 \Rightarrow \beta_1 \oplus \beta_3 \le \beta_2 \oplus \beta_3$

Property 3-14: $\beta_1 \oplus \beta_2 \le \beta_1^{\uparrow} \circ \beta_2$

4.2. Typing Rules

We adopt the same notion of judgement as [2, 3] in

our type system. $\Gamma \vdash \Diamond$ means Γ is a good environment, $\Gamma \vdash M{:}W$ means that capability M is a good expression of message type W under environment Γ, and $\Gamma \vdash P{:}T$ means P is a good process of type T under Γ. The typing rules for ROAM are listed as follows: (S_{any} means any type of message exchange)

(Env Empty) $$\frac{}{\phi \vdash \Diamond}$$

(Env n) $$\frac{\Gamma \vdash \Diamond,\ n \notin dom(\Gamma)}{\Gamma, n:W \vdash \Diamond}$$

(Cap n) $$\frac{\Gamma', n:W, \Gamma'' \vdash \Diamond}{\Gamma', n:W, \Gamma'' \vdash n:W}$$

(Cap Eps) $$\frac{\Gamma \vdash \Diamond}{\Gamma \vdash \varepsilon : Cap[S_{any}]\Omega 0}$$

(Cap in) $$\frac{\Gamma \vdash M : Amb[S_1]\alpha\beta}{\Gamma \vdash in\ M : Cap[S_{any}]\uparrow 1}$$

(Cap $\overline{in}$) $$\frac{\Gamma \vdash M : Amb[S_1]\alpha\beta}{\Gamma \vdash \overline{in}\ M : Cap[S_{any}]\Omega 1}$$

(Cap out) $$\frac{\Gamma \vdash \Diamond}{\Gamma \vdash out : Cap[S_{any}]\uparrow 1}$$

(Cap $\overline{out}$) $$\frac{\Gamma \vdash M : Amb[S_1]\alpha\beta}{\Gamma \vdash \overline{out}\ M : Cap[S_{any}]\Omega 1}$$

(Cap $open$) $$\frac{\Gamma \vdash M : Amb[S_1]\alpha\beta}{\Gamma \vdash open\ M : Cap[S_1]\alpha\beta^{\uparrow}}$$

(Cap $\overline{open}$) $$\frac{\Gamma \vdash \Diamond}{\Gamma \vdash \overline{open} : Cap[S_{any}]\Omega 1}$$

(Cap Path) $$\frac{\Gamma \vdash M_1 : Cap[S_1]\alpha_1\beta_1,\ \Gamma \vdash M_2 : Cap[S_1]\alpha_2\beta_2}{\Gamma \vdash M_1.M_2 : Cap[S_1](\alpha_1 \otimes \alpha_2)(\beta_1 \circ \beta_2)}$$

(Proc Inact) $$\frac{\Gamma \vdash \Diamond}{\Gamma \vdash \mathbf{0} : S_{any}\Omega 0}$$

(Proc Res) $$\frac{\Gamma, n:Amb[S_1]\alpha_1\beta_1 \vdash P : S_2\alpha_2\beta_2}{\Gamma \vdash (\nu n:Amb[S_1]\alpha_1\beta_1)P : S_2\alpha_2\beta_2}$$

(Proc Par) $$\frac{\Gamma \vdash P : S_1\alpha_1\beta_1,\ \Gamma \vdash Q : S_1\alpha_2\beta_2}{\Gamma \vdash P \mid Q : S_1(\alpha_1 \otimes \alpha_2)(\beta_1 \oplus \beta_2)}$$

(Proc Rep) $$\frac{\Gamma \vdash P : S_1\alpha\beta}{\Gamma \vdash !P : S_1\alpha\ (\beta \oplus \beta)}$$

(Proc Amb) $$\frac{\Gamma \vdash M : Amb[S_1]\alpha_1\beta_1,\ \Gamma \vdash P : S_1\alpha_2\beta_2,\ \alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1}{\Gamma \vdash M[P] : S_{any}\Omega 0}$$

(Proc Act) $$\frac{\Gamma \vdash M : Cap[S_1]\alpha_1\beta_1,\ \Gamma \vdash P : S_1\alpha_2\beta_2}{\Gamma \vdash M.P : S_1(\alpha_1 \otimes \alpha_2)(\beta_1 \circ \beta_2)}$$

(Proc Abs) $$\frac{\Gamma, n_1:W_1,\ \ldots, n_k:W_k \vdash P : W_1 \times \cdots \times W_k\alpha\beta}{\Gamma \vdash (n_1:W_1,\ \ldots, n_k:W_k)P : W_1 \times \cdots \times W_k\alpha\beta}$$

(Proc Msg) $$\frac{\Gamma \vdash M_1 : W_1,\ \ldots, \Gamma \vdash M_k : W_k}{\Gamma \vdash \langle M_1,\ \ldots, M_k \rangle : W_1 \times \ldots \times W_k\Omega 0}$$

The correctness of our type system is proved by the following proposition.

Proposition 3-15: *If* $\Gamma \vdash P{:}\ S_1\alpha_1\beta_1$, $P \to Q$, *then* $\Gamma \vdash Q{:}\ S_1\alpha_2\beta_2$ *and* $\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$.

The proof is in Appendix A. The result "$\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$ " show that the reduction will only cause the mobility tag and the threads count tag to decrease. This means that, in the reduction process, an immobile process will never become mobile, and the threads number in a process will never increase.

5. Examples

In this section we will show that by imposing parameter restriction, ROAM still has the strong expressiveness as its ancestors MA and SA.

5.1. Firewall-crossing

The firewall-crossing example demonstrates the ability of ambients to move between administrative domains.

Agent **::=** $k'[\overline{in}\ k.open\ k.(x{:}W).in\ x.\overline{open} \mid k''[\overline{open} \mid Q]]$

Firewall **::=** $(\nu w)w[k[out.in\ k'.\overline{open}.\langle w \rangle] \mid \overline{out}\ k.\overline{in}\ k'.open\ k'.open\ k''.P]$

Given any type for the unknown process P and Q :

$\Gamma \vdash P{:}\ S_p\alpha_p\beta_p$ and $\Gamma \vdash Q{:}\ S_q\alpha_q\beta_q$

We could type ambients *w, k, k', k"* as:

$\Gamma \vdash w$ **:** $Amb\ [S_p]\uparrow n$

$\Gamma \vdash k$ **:** $Amb[\Gamma(w)]\uparrow 1$

$\Gamma \vdash k'$**:** $Amb\ [\Gamma(w)]\uparrow n$

$\Gamma \vdash k''$ **:** $Amb\ [S_q]\alpha_q\beta_q$

As long as *k, k', k"* are kept secret, the *agent* will be authenticated and continue as *Q* in the *firewall*. That is:

$(\nu k\ k'\ k'')(Agent \mid Firewall) \approx ((\nu w)w[P \mid Q])$

The congruence relation "≈" is not introduced in this paper. But the reader could take it as the normal contextual congruence. It means that the two process will behave the same under all contextual environment. The robustness of ROAM is demonstrated again that there is no needed to forbid some special kind of capabilities to be used in the contextual environment as that is in SA.

5.2. Encoding the polyadic asynchronous π-

calculus

The encoding of the polyadic asynchronous π-calculus is an important test of the expressiveness. In the polyadic asynchronous π-calculus, a channel n of type $Ch[W_1, \ldots, W_k]$ carries n-tuples of channels, whose i-th component has type W_i. The type judgements are of the form $\Gamma \vdash P$, where Γ is the type environment and P is a π-process. The encoding in ROAM is shown below.

$[\![\Gamma \vdash P]\!] ::= [\![\Gamma]\!] \vdash [\![P]\!] : Shh \uparrow n$

$[\![\phi, n_1{:}W_1 \ldots, n_k{:}W_k]\!] ::= \phi, n_1{:}\,[\![W_1]\!] \ldots, n_k{:}\,[\![W_k]\!]$

$[\![Ch[W_1, \ldots, W_k]]\!] ::= Amb[[\![W_1]\!]\times\ldots\times[\![W_k]\!]] \uparrow n$

$[\![(\nu^{\pi} n{:}Ch[W_1, \ldots, W_k])P]\!] ::= (\nu n{:}\,[\![Ch[W_1, \ldots, W_k]]\!])\ (n[!(\overline{in}\ n.open\ n)] \mid [\![P]\!])$

$[\![n(n_1{:}W_1, \ldots, n_k{:}W_k).P]\!] ::= (\nu p{:}Amb[Shh] \uparrow n)(\overline{open}\ p \mid n[in\ n.\overline{open}.(n_1{:}\,[\![W_1]\!], \ldots, n_k{:}\,[\![W_k]\!]).(p[out.\overline{open}.\,[\![P]\!]] \mid \overline{out}\ p)])$

$[\![n\langle n_1, \ldots, n_k\rangle]\!] ::= n[in\ n.\overline{open}.\langle n_1, \ldots, n_k\rangle]$

$[\![P \mid Q]\!] ::= [\![P]\!] \mid [\![Q]\!]$

$[\![!P]\!] ::= ![\![P]\!]$

$[\![\mathbf{0}]\!] ::= \mathbf{0}$

In the encoding, a channel n is represented as an ambient n, which continuously allows the communication-carrying ambients to enter and opens them. Communications along channel n become message exchanges within ambient n. The input and output actions are encoded as ambients that jump into n and be opened there. After that, the message exchange happens and the receiver jumps out n and continues.

In this version of encoding, only one ambient name is used for every channel. In the mean time, two ambients with the same name act as the input-carrying ambient and the output-carrying ambient respectively. We could also use two different names for them. But as long as no confusing is raised, it is simpler and clearer to use just one ambient name for every channel.

6. Conclusion and Future Works

In this paper, a more robust version of the ambient calculus – ROAM is proposed as a more rational substitute for SA to control the grave interference in MA. A new type system with the mobility tag and the threads count tag are presented and its correctness proved. The encoding of the firewall-crossing example and the polyadic asynchronous π-calculus show that ROAM also inherits the strong expressiveness of its ancestors.

Although ROAM has some ideal properties than its ancestors, the study of ROAM in this paper is just a preliminary step. There are still many problems to be solved, among which are the reduction properties of the single-threaded process, the type notion for the degrading of both the mobility and threads count, etc. We will continue to explore them with great enthusiasm.

References

1. L. Cardelli, A. D. Gordon. Mobile Ambients. *Foundations of Software Science and Computational Structures, Lecture Notes in Computer Science*, No. 1378, Springer Verlag (D), pp. 140-155, 1998. Also available from : http://www.luca.demon.co.uk/
2. L. Cardelli and A. D. Gordon. Types for Mobile Ambients. *Proc. 26th POPL*, pages 79-92. ACM Press. 1999.
3. L. Cardelli, G. Ghelli, and A. D. Gordon. Mobility Types for Mobile Ambients. In *Proc. ICALP'99, Lecture Notes in Computer Science*, No. 1644, Springer Verlag (D), pp. 230-239, 1999.
4. F. Levi and D. Sangiorgi. Controlling Interference in Ambients. To appear in *Proc. 27th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, Boston, Massachusetts, Jan. 19-21, 2000. Also available at: ftp://zenon.inria.fr/meije/theorie-par/davides/ SafeAmbients.ps
5. R. Milner. The polyadic π-calculus: a tutorial. In *Logic and Algebra of Specification*. Springer Verlag, 1993.

Appendix A. The Proof of Proposition 3-15

The proofs for the follow Lemma A-1 to Lemma A-10 are omitted due to limited paper length.

Lemma A-1: *If* $\Gamma \vdash M.P{:}S_1\alpha_1\beta_1$, *then* $\Gamma \vdash P{:}S_1\alpha_2\beta_2 \wedge\ \alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$.

Lemma A-2: *If* $\Gamma \vdash P_1{:}S_1\alpha_1\beta_1$, $\Gamma \vdash P_2{:}S_2\alpha_2\beta_2$, $\Gamma \vdash P_1|Q : S_1\alpha_3\beta_3$, *and* $\Gamma \vdash P_2|Q : S_2\alpha_4\beta_4$ *with* $\alpha_2 \le \alpha_1$ *and* $\beta_2 \le \beta_1$, *then* $\alpha_4 \le \alpha_3$ *and* $\beta_4 \le \beta_3$.

Corollary A-3: *If* $\Gamma \vdash M.P|Q{:}S_1\alpha_1\beta_1$, *then* $\Gamma \vdash P|Q{:}S_1\alpha_2\beta_2 \wedge \alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$.

Let $\Gamma \vdash J$ denote any judgement.

Lemma A-4: *If* $\Gamma', n{:}W, \Gamma'' \vdash J$ *then* $n \notin dom(\Gamma', \Gamma'')$.

Lemma A-5: *If* $\Gamma \vdash n{:}W$ *and* $\Gamma \vdash n{:}W'$, *then* $W = W'$.

Lemma A-6 (Implied Judgement): *If* $\Gamma', \Gamma'' \vdash J$ *then* $\Gamma' \vdash \Diamond$.

Lemma A-7 (Exchange): *If* $\Gamma', n{:}W', m{:}W'', \Gamma'' \vdash J$ *then* $\Gamma',$

$m{:}W''$, $n{:}W'$, $\Gamma'' \vdash J$.

Lemma A-8 (Weakening): *If* Γ', $\Gamma'' \vdash J$ *and* $n \notin dom(\Gamma', \Gamma'')$ *then* Γ', $n{:}W$, $\Gamma'' \vdash J$.

Lemma A-9 (Strengthening): *If* Γ', $n{:}W$, $\Gamma'' \vdash J$ *and* $n \notin fn(J)$ *then* Γ', $\Gamma'' \vdash J$.

Lemma A-10 (Substituition): *If* $\Gamma' \vdash M'{:}W$ *and* Γ', $n{:}W$, $\Gamma'' \vdash J$, *then* Γ', $\Gamma'' \vdash J\{M'/n\}$.

Lemma A-11 (Subject Congruence):
(1) *If* $\Gamma \vdash P{:}S_1\alpha_1\beta_1$ *and* $P \equiv Q$, *then* $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.
(2) *If* $\Gamma \vdash P{:}S_1\alpha_1\beta_1$ *and* $Q \equiv P$, *then* $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.
Proof: By mutual induction on the derivation of $P \equiv Q$ and $Q \equiv P$.
(1) *If* $\Gamma \vdash P{:}S_1\alpha_1\beta_1$ *and* $P \equiv Q$, *then* $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.
(Struct Ref) Trival
(Struct Symm) Then $Q \equiv P$, By induction hypothesis (2), we have $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.
(Struct Trans) Then $P \equiv R$, $R \equiv Q$ for some R. By induction hypothesis (1), $\Gamma \vdash R :S_1\alpha_1\beta_1$. Again by induction hypothesis (1), $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.
(Struct Res) Then $P=(\nu n{:}W)P'$ and $Q=(\nu n{:}W)Q'$, with $P' \equiv Q'$. Assume $\Gamma \vdash (\nu n{:}W)P' :S_1\alpha_1\beta_1$. This must have been derived from (Proc Res), with Γ, $n{:}W \vdash P' :S_1\alpha_1\beta_1$. By induction hypothesis (1), Γ, $n{:}W \vdash Q' :S_1\alpha_1\beta_1$. By (Proc Res), $\Gamma \vdash (\nu n{:}W)Q' :S_1\alpha_1\beta_1$.
(Struct Par) Then $P=P'|R$ and $Q=Q'|R$, with $P' \equiv Q'$. Assume $\Gamma \vdash P'|R :S_1\alpha_1\beta_1$. This must have been derived from (Proc Par), with $\Gamma \vdash P':S_1\alpha_2\beta_2$, $\Gamma \vdash R{:}S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \oplus \beta_3$. By induction hypothesis (1), $\Gamma \vdash Q' :S_1\alpha_2\beta_2$. By (Proc Par), $\Gamma \vdash Q'|R :S_1\alpha_1\beta_1$.
(Struct Rep) Then $P=!P'$ and $Q=!Q'$, with $P' \equiv Q'$. Assume $\Gamma \vdash !P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Rep), with $\Gamma \vdash P':S_1\alpha_1\beta_2$ and $\beta_1=\beta_2 \oplus \beta_2$. By induction hypothesis (1), $\Gamma \vdash Q' :S_1\alpha_1\beta_2$. By (Proc Rep), $\Gamma \vdash !Q' :S_1\alpha_1\beta_1$.
(Struct Amb) Then $P=M[P']$ and $Q=M[Q']$, with $P' \equiv Q'$. Assume $\Gamma \vdash M[P']{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Amb), with $\Gamma \vdash M{:}Amb[S_2]\alpha_2\beta_2$, $\Gamma \vdash P':S_2\alpha_3\beta_3$, $\alpha_3 \le \alpha_2 \wedge \beta_3 \le \beta_2$ and $\alpha_1=\Omega \wedge \beta_1=0$. By induction hypothesis (1), $\Gamma \vdash Q' :S_2\alpha_3\beta_3$. By (Proc Amb), $\Gamma \vdash M[Q'] :S_1\alpha_1\beta_1$.
(Struct Act) Then $P=M.P'$ and $Q=M.Q'$, with $P' \equiv Q'$. Assume $\Gamma \vdash M.P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Act), with $\Gamma \vdash M{:}Cap[S_1]\alpha_2\beta_2$, $\Gamma \vdash P':S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \circ \beta_3$. By induction hypothesis (1), $\Gamma \vdash Q' :S_1\alpha_3\beta_3$. By (Proc Act), $\Gamma \vdash M.Q':S_1\alpha_1\beta_1$.
(Struct Abs) Then $P=(n_1{:}W_1, \ldots, n_k{:}W_k).P'$ and $Q=(n_1{:}W_1, \ldots, n_k{:}W_k).Q'$, with $P' \equiv Q'$. Assume $\Gamma \vdash (n_1{:}W_1, \ldots, n_k{:}W_k).P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Abs), with Γ, $n_1{:}W_1, \ldots, n_k{:}W_k \vdash P':W_1 \times \ldots \times W_k\ \alpha_1\beta_1$ and $S_1=W_1 \times \ldots \times W_k$. By induction hypothesis (1), Γ, $n_1{:}W_1, \ldots, n_k{:}W_k \vdash Q':W_1 \times \ldots \times W_k\ \alpha_1\beta_1$. By (Proc Abs), $\Gamma \vdash (n_1{:}W_1, \ldots, n_k{:}W_k).Q':S_1\alpha_1\beta_1$.
(Struct Par Comm) Then $P=P'|P''$ and $Q=P''|P'$. Assume $\Gamma \vdash P'|P'' :S_1\alpha_1\beta_1$. This must have been derived from (Proc Par), with $\Gamma \vdash P':S_1\alpha_2\beta_2$, $\Gamma \vdash P'':S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \oplus \beta_3$. By the symmetry of operators $\otimes$ and $\oplus$, we have $\alpha_1=\alpha_3 \otimes \alpha_2 \wedge \beta_1=\beta_3 \oplus \beta_2$. By (Proc Par), $\Gamma \vdash P''|P' :S_1\alpha_1\beta_1$.
(Struct Par Assoc) Then $P=(P'|P'')|P'''$ and $Q=P'|(P''|P''')$. Assume $\Gamma \vdash (P'|P'')|P'''{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Par) twice, with $\Gamma \vdash P':S_1\alpha_2\beta_2$, $\Gamma \vdash P'':S_1\alpha_3\beta_3$, $\Gamma \vdash P'''{:}S_1\alpha_4\beta_4$ and $\alpha_1=(\alpha_2 \otimes \alpha_3) \otimes \alpha_4 \wedge \beta_1=(\beta_2 \oplus \beta_3) \oplus \beta_4$. By the associativity of operators $\otimes$ and $\oplus$, we have $\alpha_1=\alpha_2 \otimes (\alpha_3 \otimes \alpha_4) \wedge \beta_1=\beta_2 \oplus (\beta_3 \oplus \beta_4)$. By (Proc Par) twice, $\Gamma \vdash P'|(P''|P''') : S_1\alpha_1\beta_1$.
(Struct Rep Par) Then $P=!P'$ and $Q=P'|!P'$. Assume $\Gamma \vdash !P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Rep), with $\Gamma \vdash P':S_1\alpha_1\beta_2$ and $\beta_1=\beta_2 \oplus \beta_2$. By property 3-11, we have $\beta_1=\beta_2 \oplus (\beta_2 \oplus \beta_2)=\beta_2 \oplus \beta_1$. By (Proc Par), $\Gamma \vdash P'|!P' :S_1\alpha_1\beta_1$.
(Struct Res Res) Then $P=(\nu n{:}W_1)(\nu m{:}W_2)P'$ and $Q=(\nu m{:}W_2)(\nu n{:}W_1)P'$. Assume $\Gamma \vdash (\nu n{:}W_1)(\nu m{:}W_2)P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Res) twice, with Γ, $n{:}W_1$, $m{:}W_2 \vdash P':S_1\alpha_1\beta_1$. By Lemma A-7, we have Γ, $m{:}W_2$, $n{:}W_1 \vdash P':S_1\alpha_1\beta_1$. By (Proc Res) twice, $\Gamma \vdash (\nu m{:}W_2)(\nu n{:}W_1)P' : S_1\alpha_1\beta_1$.
(Struct Res Par) Then $P=(\nu n{:}W)(P'|P'')$ and $Q=P'|(\nu n{:}W)P''$ with $n \notin fn(P')$. Assume $\Gamma \vdash (\nu n{:}W)(P'|P''):S_1\alpha_1\beta_1$. This must have been derived from (Proc Res), with Γ, $n{:}W \vdash P'|P'':S_1\alpha_1\beta_1$ and from (Proc Par) with Γ, $n{:}W \vdash P':S_1\alpha_2\beta_2$, Γ, $n{:}W \vdash P'':S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \oplus \beta_3$. By (Proc Res) we have $\Gamma \vdash (\nu n{:}W)P'':S_1\alpha_3\beta_3$. By Lemma A-9, since $n \notin fn(P')$, we have $\Gamma \vdash P':S_1\alpha_2\beta_2$. By (Proc Par), we have $\Gamma \vdash P'|(\nu n{:}W)P'':S_1\alpha_1\beta_1$.
(Struct Res Amb) Then $P=(\nu n{:}W)m[P']$ and $Q= m[(\nu n{:}W)P']$ with $n \neq m$. Assume $\Gamma \vdash (\nu n{:}W)m[P']:S_1\alpha_1\beta_1$. This must have been derived from (Proc Res), with Γ, $n{:}W \vdash m[P']:S_1\alpha_1\beta_1$ and from (Proc Amb) with Γ, $n{:}W \vdash m{:}Amb[S_2]\alpha_2\beta_2$, Γ, $n{:}W \vdash P':S_2\alpha_3\beta_3$, $\alpha_3 \le \alpha_2 \wedge \beta_3 \le \beta_2$ and $\alpha_1=\Omega \wedge \beta_1=0$. By Lemma A-9, since $n \neq m$, we have $\Gamma \vdash m{:}Amb[S_2]\alpha_2\beta_2$, By (Proc Res), we have $\Gamma \vdash (\nu n{:}W)P':S_2\alpha_3\beta_3$. By (Proc Amb), we have $\Gamma \vdash m[(\nu n{:}W)P']:S_1\alpha_1\beta_1$.
(Struct Zero Par) Then $P=P'|\mathbf{0}$ and $Q=P'$. Assume $\Gamma \vdash P'|\mathbf{0} :S_1\alpha_1\beta_1$. This must have been derived from (Proc Par), with $\Gamma \vdash P':S_1\alpha_2\beta_2$, $\Gamma \vdash \mathbf{0}{:}S_1\Omega 0$ and $\alpha_1=\alpha_2 \otimes \Omega \wedge \beta_1=\beta_2 \oplus 0$. By the definition of operators $\otimes$ and $\oplus$, we have $\alpha_1=\alpha_2 \wedge \beta_1=\beta_2$, or $\Gamma \vdash P':S_1\alpha_1\beta_1$.
(Struct Zero Res) Then $P=(\nu n{:}W)\mathbf{0}$ and $Q=\mathbf{0}$. Assume $\Gamma \vdash (\nu n{:}W)\mathbf{0}:S_1\alpha_1\beta_1$. This must have been derived from (Proc Res), with Γ, $n{:}W \vdash \mathbf{0}:S_1\alpha_1\beta_1$. By Lemma A-9,we have $\Gamma \vdash \mathbf{0}:S_1\alpha_1\beta_1$.
(Struct Zero Rep) Then $P=!\mathbf{0}$ and $Q=\mathbf{0}$. Assume $\Gamma \vdash !\mathbf{0}:S_1\alpha_1\beta_1$. This must have been derived from (Proc Rep), with $\Gamma \vdash \mathbf{0}:S_1\alpha_1\beta_2$ and $\beta_1=\beta_2 \oplus \beta_2$ and from (Proc Inact) with $\alpha_1=\Omega \wedge \beta_2=0$. So $\beta_1=0 \oplus 0=0=\beta_2$.
(Struct Eps) Then $P=\varepsilon.P'$ and $Q=P'$. Assume $\Gamma \vdash \varepsilon.P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Act), with $\Gamma \vdash \varepsilon{:}Cap[S_1]\alpha_2\beta_2$, $\Gamma \vdash P':S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \circ \beta_3$. The former must come from (Cap Eps) with $\alpha_2=\Omega \wedge \beta_2=0$. By the definition of operators $\otimes$ and $\circ$, we have $\alpha_1=\alpha_3 \wedge \beta_1=\beta_3$, or $\Gamma \vdash P':S_1\alpha_1\beta_1$.
(Struct Path) Then $P=(M.M').P'$ and $Q=M.(M'.P')$. Assume $\Gamma \vdash (M.M').P':S_1\alpha_1\beta_1$. This must have been derived from (Proc Act), with $\Gamma \vdash M.M'{:}Cap[S_1]\alpha_2\beta_2$, $\Gamma \vdash P':S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2 \otimes \alpha_3 \wedge \beta_1=\beta_2 \circ \beta_3$. The former must come from (Cap Path) with $\Gamma \vdash M{:}Cap[S_1]\alpha_4\beta_4$, $\Gamma \vdash M'{:}Cap[S_1]\alpha_5\beta_5$ and $\alpha_2=\alpha_4 \otimes \alpha_5 \wedge \beta_2=\beta_4 \circ \beta_5$. By property 3-2 and 3-6, we have $\alpha_1=(\alpha_4 \otimes \alpha_5) \otimes \alpha_3=\alpha_4 \otimes (\alpha_5 \otimes \alpha_3) \wedge \beta_1=(\beta_4 \circ \beta_5) \circ \beta_3=\beta_4 \circ (\beta_5 \circ \beta_3)$. By (Proc Act) twice, we have $\Gamma \vdash M.(M'.P'):S_1\alpha_1\beta_1$.

(2) *If* $\Gamma \vdash P{:}S_1\alpha_1\beta_1$ *and* $Q \equiv P$, *then* $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.

(Struct Ref) Trival

(Struct Symm) Then $P \equiv Q$, By induction hypothesis (1), we have $\Gamma \vdash Q{:}S_1\alpha_1\beta_1$.

(Struct Trans) (Struct Res) (Struct Par) (Struct Rep) (Struct Amb) (Struct Act) (Struct Abs) (Struct Par Comm) (Struct Par Assoc) Symmetrical/analogy to case (1).

(Struct Rep Par) Then $Q=!P'$ and $P=P'|!P'$. Assume $\Gamma \vdash P'|!P'{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Par), with $\Gamma \vdash P'{:}S_1\alpha_2\beta_2$, $\Gamma \vdash !P'{:}S_1\alpha_3\beta_3$, and $\alpha_1=\alpha_2\otimes\alpha_3 \wedge \beta_1=\beta_2\oplus\beta_3$. The later must have been derived from (Proc Rep), so $\alpha_2=\alpha_3 \wedge \beta_3=\beta_2\oplus\beta_2$. By property 3-3, we have $\alpha_1=\alpha_3\otimes\alpha_3=\alpha_3$. By property 3-11, we have $\beta_1=\beta_2\oplus(\beta_2\oplus\beta_2)=\beta_2\oplus\beta_2=\beta_3$. So we have $\Gamma \vdash !P'{:}S_1\alpha_1\beta_1$.

(Struct Res Res) Symmetrical to case (1).

(Struct Res Par) Then $Q=(\nu n{:}W)(P'|P'')$ and $P=P'|(\nu n{:}W)P''$ with $n \notin fn(P')$. Assume $\Gamma \vdash P'|(\nu n{:}W)P''{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Par), with $\Gamma \vdash P'{:}S_1\alpha_2\beta_2$, $\Gamma \vdash (\nu n{:}W)P''{:}S_1\alpha_3\beta_3$ and $\alpha_1=\alpha_2\otimes\alpha_3 \wedge \beta_1=\beta_2\oplus\beta_3$. The later must have been derived from (Proc Res) with $\Gamma, n{:}W \vdash P''{:}S_1\alpha_3\beta_3$. By Lemma A-8, since $n \notin fn(P')$, we have $\Gamma, n{:}W \vdash P'{:}S_1\alpha_2\beta_2$. By (Proc Par) and (Proc Res), we obtain $\Gamma \vdash (\nu n{:}W)(P'|P''){:}S_1\alpha_1\beta_1$.

(Struct Res Amb) Then $Q=(\nu n{:}W)m[P']$ and $P=m[(\nu n{:}W)P']$ with $n \neq m$. Assume $\Gamma \vdash m[(\nu n{:}W)P']{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Amb), with $\Gamma \vdash m{:}Amb[S_2]\alpha_2\beta_2$, $\Gamma \vdash (\nu n{:}W)P'{:}S_2\alpha_3\beta_3$, $\alpha_3 \le \alpha_2 \wedge \beta_3 \le \beta_2$ and $\alpha_1=\Omega \wedge \beta_1=0$. The later must have been derived from (Proc Res) with $\Gamma, n{:}W \vdash P'{:}S_2\alpha_3\beta_3$. By Lemma A-8, since $n \neq m$, we have $\Gamma, n{:}W \vdash m{:}Amb[S_2]\alpha_2\beta_2$, By (Proc Amb), we have $\Gamma, n{:}W \vdash m[P']{:}S_1\alpha_1\beta_1$. By (Proc Res), we have $\Gamma \vdash (\nu n{:}W)m[P']{:}S_1\alpha_1\beta_1$.

(Struct Zero Par) Then $Q=P'|\mathbf{0}$ and $P=P'$. Assume $\Gamma \vdash P'{:}S_1\alpha_1\beta_1$. By Lemma A-6 we have $\Gamma \vdash \Diamond$. By (Proc Inact) we have $\Gamma \vdash \mathbf{0}{:}S_1\Omega 0$. By (Proc Par), $\Gamma \vdash P'|\mathbf{0}{:}S_1(\alpha_1\otimes\Omega)\ (\beta_1\oplus 0)$. By the definition of operators $\otimes$ and $\oplus$, we have $\Gamma \vdash P'|\mathbf{0}{:}S_1\alpha_1\beta_1$.

(Struct Zero Res) Then $Q=(\nu n{:}W)\mathbf{0}$ and $P=\mathbf{0}$. Assume $\Gamma \vdash \mathbf{0}{:}S_1\alpha_1\beta_1$(here we assume $n \notin dom(\Gamma)$ by renaming). By Lemma A-8,we have $\Gamma, n{:}W \vdash \mathbf{0}{:}S_1\alpha_1\beta_1$. By (Proc Res), we have $\Gamma \vdash (\nu n{:}W)\mathbf{0}{:}S_1\alpha_1\beta_1$.

(Struct Zero Rep) Then $Q=!\mathbf{0}$ and $P=\mathbf{0}$. Assume $\Gamma \vdash \mathbf{0}{:}S_1\alpha_1\beta_1$. This must have been derived from (Proc Inact) with $\alpha_1=\Omega \wedge \beta_1=0$. By (Proc Rep), we have $\Gamma \vdash !\mathbf{0}{:}S_1\alpha_1\beta_2$ and $\beta_2=\beta_1\oplus\beta_1=0\oplus 0=0=\beta_1$. So we have $\Gamma \vdash !\mathbf{0}{:}S_1\alpha_1\beta_1$.

(Struct Eps) Then $Q=\boldsymbol{\varepsilon}.P'$ and $P=P'$. Assume $\Gamma \vdash P'{:}S_1\alpha_1\beta_1$. By Lemma A-6 we have $\Gamma \vdash \Diamond$. By (Proc Eps) we have $\Gamma \vdash \boldsymbol{\varepsilon}{:}Cap[S_1]\Omega 0$. By (Proc Act) we have $\Gamma \vdash \boldsymbol{\varepsilon}.P'{:}S_1(\alpha_1\otimes\Omega)\ (\beta_1 \circ 0)$. By the definition of operators $\otimes$ and $\circ$, we have $\Gamma \vdash \boldsymbol{\varepsilon}.P'{:}S_1\alpha_1\beta_1$.

(Struct Path) Analog to case (1). □

Proposition 3-15: *If* $\Gamma \vdash P{:}\ S_1\alpha_1\beta_1$, $P \to Q$, *then* $\Gamma \vdash Q{:}\ S_1\alpha_2\beta_2$ *and* $\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$.

Proof: By induction on the derivation of $P \to Q$.

(R-*in*):

(1) $\Gamma \vdash n[\overline{in}\ m.P_1|P_2\,]|m[in\ n.Q_1|Q_2]{:}S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash n[\overline{in}\ m.P_1|P_2\,]{:}S_1\alpha_2\beta_2$ (1)+(Proc Par) only

(3) $\Gamma \vdash m[in\ n.Q_1|Q_2]{:}S_1\alpha_3\beta_3$ along with (2)

(4) $\alpha_1=\alpha_2\otimes\alpha_3,\ \beta_1=\beta_2\oplus\beta_3$ along with (2)

(5) $\alpha_2=\Omega \wedge \beta_2=0$ (2)+(Proc Amb) only

(6) $\Gamma \vdash n{:}Amb[S_n]\alpha_n\beta_n$ along with (5)

(7) $\Gamma \vdash \overline{in}\ m.P_1|P_2{:}S_n\alpha_4\beta_4$ along with (5)

(8) $\alpha_4 \le \alpha_n \wedge \beta_4 \le \beta_n$ along with (5)

(9) $\alpha_3=\Omega \wedge \beta_3=0$ (3)+(Proc Amb) only

(10) $\Gamma \vdash m{:}Amb[S_m]\alpha_m\beta_m$ along with (9)

(11) $\Gamma \vdash in\ n.Q_1|Q_2{:}S_m\alpha_5\beta_5$ along with (9)

(12) $\alpha_5 \le \alpha_m \wedge \beta_5 \le \beta_m$ along with (9)

(13) $\Gamma \vdash P_1|P_2{:}S_n\alpha_6\beta_6$ (7)+ Corollary A-3

(14) $\alpha_6 \le \alpha_4 \wedge \beta_6 \le \beta_4$ along with (13)

(15) $\Gamma \vdash Q_1|Q_2{:}S_m\alpha_7\beta_7$ (11)+Corollary A-3

(16) $\alpha_7 \le \alpha_5 \wedge \beta_7 \le \beta_5$ along with (15)

(17) $\alpha_6 \le \alpha_n \wedge \beta_6 \le \beta_n$ (8)(14)+ trans. of $\le$

(18) $\alpha_7 \le \alpha_m \wedge \beta_7 \le \beta_m$ (12)(16)+ trans. of $\le$

(19) $\Gamma \vdash m[Q_1|Q_2]{:}S\Omega 0$ (10)(15)(18)+(Proc Amb)

(20) $\Gamma \vdash P_1|P_2|m[Q_1|Q_2]{:}S_n\alpha_6\beta_6$ (13)(19)+(Proc Par)

(21) $\Gamma \vdash n[P_1|P_2|m[Q_1|Q_2]]{:}S_1\Omega 0$ (6)(17)(20)+(Proc Amb)

(22) $\alpha_1=\Omega \wedge \beta_1=0$ (4)(5)(9)+Def $\otimes,\oplus$

From (21), (22), we know the proposition is true under (R-*in*).

(R-*out*):

(1) $\Gamma \vdash n[\overline{out}\ m.P_1|P_2|m[out.Q_1|Q_2]]{:}S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash n{:}Amb[S_n]\alpha_n\beta_n$ (1)+(Proc Amb) only

(3) $\Gamma \vdash \overline{out}\ m.P_1|P_2|m[out.Q_1|Q_2]{:}S_n\alpha_2\beta_2$ along with (2)

(4) $\alpha_1=\Omega,\ \beta_1=0$ along with (2)

(5) $\alpha_2 \le \alpha_n \wedge \beta_2 \le \beta_n$ along with (2)

(6) $\Gamma \vdash \overline{out}\ m.P_1|P_2{:}S_n\alpha_3\beta_3$ (3)+(Proc Par) only

(7) $\alpha_3 \le \alpha_2 \wedge \beta_3 \le \beta_2$ along with (6)+property 3-4,3-12

(8) $\Gamma \vdash m[out.Q_1|Q_2]{:}S_n\alpha_4\beta_4$ along with (6)

(9) $\alpha_4 \le \alpha_2 \wedge \beta_4 \le \beta_2$ along with (6) + property 3-4,3-12

(10) $\Gamma \vdash P_1|P_2{:}S_n\alpha_5\beta_5$ (6)+Corollary A-3

(11) $\alpha_5 \le \alpha_3 \wedge \beta_5 \le \beta_3$ along with (10)

(12) $\alpha_5 \le \alpha_n \wedge \beta_5 \le \beta_n$ (5)(7)(11)+ trans. of $\le$

(13) $\Gamma \vdash n[P_1|P_2]{:}\ S_{any}\Omega 0$ (2)(10)(12)+(Proc Amb)

(14) $\Gamma \vdash m{:}Amb[S_m]\alpha_m\beta_m$ (8)+(Proc Amb) only

(15) $\Gamma \vdash out.Q_1|Q_2{:}S_m\alpha_6\beta_6$ along with (14)

(16) $\alpha_6 \le \alpha_m \wedge \beta_6 \le \beta_m$ along with (14)

(17) $\alpha_4=\Omega,\ \beta_4=0$ along with (14)

(18) $\Gamma \vdash Q_1|Q_2{:}S_m\alpha_7\beta_7$ (15)+Corollary A-3

(19) $\alpha_7 \le \alpha_6 \wedge \beta_7 \le \beta_6$ along with (18)

(20) $\alpha_7 \le \alpha_m \wedge \beta_7 \le \beta_m$ (16)(19)+ trans. of $\le$

(21) $\Gamma \vdash m[Q_1|Q_2]{:}\ S_{any}\Omega 0$ (14)(18)(20)+(Proc Amb)

(22) $\Gamma \vdash n[P_1|P_2]|m[Q_1|Q_2]{:}\ S_{any}\Omega 0$ (13)(21)+(Proc Par)

From (4), (22), we know the proposition is true under (R-*out*).

(R-*open*):

(1) $\Gamma \vdash open\ n.P|n[\overline{open}\ .Q_1|Q_2]{:}S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash open\ n.P:S_1\alpha_2\beta_2$ (1)+(Proc Par) only

(3) $\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$ along with (2)+property 3-4, 3-12

(4) $\Gamma \vdash n[\overline{open}.Q_1|Q_2]:S_1\alpha_3\beta_3$ along with (2)

(5) $\alpha_3 \le \alpha_1 \wedge \beta_3 \le \beta_1$ along with (2)+property 3-4, 3-12

(6) $\Gamma \vdash n:Amb[S_n]\alpha_n\beta_n$ (4)+(Proc Amb) only

(7) $\Gamma \vdash \overline{open}.Q_1|Q_2:S_n\alpha_4\beta_4$ along with (6)

(8) $\alpha_4 \le \alpha_n \wedge \beta_4 \le \beta_n$ along with (6)

(9) $\Gamma \vdash Q_1|Q_2:S_n\alpha_5\beta_5$ (7)+Corollary A-3

(10) $\alpha_5 \le \alpha_4 \wedge \beta_5 \le \beta_4$ along with (9)

(11) $\Gamma \vdash open\ n:Cap[S_n]\alpha_n\beta_n^{\uparrow}$ (6)+(Cap *open*)

(12) $\Gamma \vdash open\ n:Cap[S_1]\alpha_6\beta_6$ (2)+(Proc Act) only

(13) $\Gamma \vdash P:S_1\alpha_7\beta_7$ along with (12)

(14) $\alpha_6 \otimes \alpha_7 = \alpha_2 \wedge \beta_6 \circ \beta_7 = \beta_2$ along with (12)

(15) $S_n = S_1 \wedge \alpha_n = \alpha_6 \wedge \beta_n^{\uparrow} = \beta_6$ (11)(12)

(16) $\alpha_n \otimes \alpha_7 = \alpha_2 \wedge \beta_n^{\uparrow} \circ \beta_7 = \beta_2$ (14)(15)

(17) $\Gamma \vdash P|Q_1|Q_2:S_1\alpha_8\beta_8$ (13)(9)(15)+(Proc Par)

(18) $\alpha_7 \otimes \alpha_5 = \alpha_8 \wedge \beta_7 \oplus \beta_5 = \beta_8$ along with (17)

(19) $\alpha_5 \le \alpha_n$ (10)(8)+ trans. of ≤

(20) $\alpha_5 \otimes \alpha_7 \le \alpha_n \otimes \alpha_7$ (19)+Property 3-5

(21) $\alpha_8 \le \alpha_2$ (18)+Property 3-1, (16)+ trans. of ≤

(22) $\alpha_8 \le \alpha_1$ (21)(3) + trans. of ≤

(23) $\beta_n \oplus \beta_7 \le \beta_2$ (16)+Property 3-14

(24) $\beta_5 \le \beta_n$ (10)(8) + trans. of ≤

(25) $\beta_5 \oplus \beta_7 \le \beta_n \oplus \beta_7$ (24)+Property 3-13

(26) $\beta_8 \le \beta_2$ (18)+Property 3-9, (25)(25)+ trans. of ≤

(27) $\beta_8 \le \beta_1$ (26)(3) + trans. of ≤

From (17), (21), (27), we know the proposition is true under (R-*open*).

(R-Msg):

(1) $\Gamma \vdash \langle M_1, \ldots, M_k\rangle|(n_1:W_1, \ldots, n_k:W_k).P:S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash \langle M_1, \ldots, M_k\rangle:S_1\alpha_2\beta_2$ (1)+(Proc Par) only

(3) $\Gamma \vdash (n_1:W_1, \ldots, n_k:W_k).P:S_1\alpha_3\beta_3$ along with (2)

(4) $\alpha_2 \otimes \alpha_3 = \alpha_1 \wedge \beta_2 \oplus \beta_3 = \beta_1$ along with (2)

(5) $\Gamma, n_1:W_1, \ldots, n_k:W_k \vdash P:S_1\alpha_3\beta_3$ (3)+(Proc Abs) only

(6) $S_1 = W_1 \times \ldots \times W_k$ along with (5)

(7) $\Gamma \vdash M_1:W_1, \ldots, \Gamma \vdash M_k:W_k$ (2)(6)+(Proc Msg) only

(8) $\Gamma \vdash P\{M_1/n_1, \ldots, M_k/n_k\}:S_1\alpha_3\beta_3$ (5)(7)+Lemma A-10 *k* times

(9) $\alpha_3 \le \alpha_1 \wedge \beta_3 \le \beta_1$ (4)+Property 3-4, 3-12

From (8), (9) we know the proposition is true under (R-Msg).

(R-Res):

(1) $\Gamma \vdash (\nu n:W)P:S_1\alpha_1\beta_1$ condition given

(2) $\Gamma, n:W \vdash P:S_1\alpha_1\beta_1$ (1)+(Proc Res) only

(3) $P \rightarrow P'$ condition given

(4) $\Gamma, n:W \vdash P':S_1\alpha_2\beta_2$ (2)(3)+induction hypothesis

(5) $\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$ along with (4)

(6) $\Gamma \vdash (\nu n:W)P':S_1\alpha_2\beta_2$ (4)+(Proc Res)

From (6), (5), we know the proposition is true under (R-Res).

(R-Par):

(1) $\Gamma \vdash P|Q:S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash P:S_1\alpha_2\beta_2$ (1)+(Proc Par) only

(3) $\Gamma \vdash Q:S_1\alpha_3\beta_3$ along with (2)

(4) $\alpha_2 \otimes \alpha_3 = \alpha_1 \wedge \beta_2 \oplus \beta_3 = \beta_1$ along with (2)

(5) $P \rightarrow P'$ condition given

(6) $\Gamma \vdash P':S_1\alpha_4\beta_4$ (2)(5)+induction hypothesis

(7) $\alpha_4 \le \alpha_2 \wedge \beta_4 \le \beta_2$ along with (6)

(8) $\Gamma \vdash P'|Q:S_1\alpha_5\beta_5$ (6)(3)+(Proc Par)

(9) $\alpha_4 \otimes \alpha_3 = \alpha_5 \wedge \beta_4 \oplus \beta_3 = \beta_5$ along with (8)

(10) $\alpha_4 \otimes \alpha_3 \le \alpha_2 \otimes \alpha_3 \wedge \beta_4 \oplus \beta_3 \le \beta_2 \oplus \beta_3$ (7)+Property 3-5, 3-13

(11) $\alpha_5 \le \alpha_1 \wedge \beta_5 \le \beta_1$ (9)(10)(4)+Property 3-4, 3-12+trans. of ≤

From (8), (11), we know the proposition is true under (R-Par).

(R-Amb):

(1) $\Gamma \vdash n[P]:S_1\alpha_1\beta_1$ condition given

(2) $\Gamma \vdash n:Amb[S_2]\alpha_2\beta_2$ (1)+(Proc Amb) only

(3) $\Gamma \vdash P:S_2\alpha_3\beta_3$ along with (2)

(4) $\alpha_3 \le \alpha_2 \wedge \beta_3 \le \beta_2$ along with (2)

(5) $\alpha_1 = \Omega$， $\beta_1 = 0$ along with (2)

(6) $P \rightarrow P'$ condition given

(7) $\Gamma \vdash P':S_1\alpha_4\beta_4$ (3)(6)+induction hypothesis

(8) $\alpha_4 \le \alpha_3 \wedge \beta_4 \le \beta_3$ along with (7)

(9) $\alpha_4 \le \alpha_2 \wedge \beta_4 \le \beta_2$ (8)(4)+ trans of ≤

(10) $\Gamma \vdash n[P']:S_1\alpha_5\beta_5$ (2)(7)(9)+(Proc Amb), let $S_{any}=S_1$

(11) $\alpha_5 = \Omega$, $\beta_5 = 0$ along with (10)

From (10), (11), (5) we know the proposition is true under (R-Amb).

(R-≡):

(1) $\Gamma \vdash P:S_1\alpha_1\beta_1$ condition given

(2) $P \equiv P'$ condition given

(3) $\Gamma \vdash P':S_1\alpha_1\beta_1$ (1)(2)+Lemma A-11

(4) $P' \rightarrow P''$ condition given

(5) $\Gamma \vdash P'':S_1\alpha_2\beta_2$ (3)(4)+induction hypothesis

(6) $\alpha_2 \le \alpha_1 \wedge \beta_2 \le \beta_1$ along with (5)

(7) $P'' \equiv P'''$ condition given

(8) $\Gamma \vdash P''':S_1\alpha_2\beta_2$ (5)(7)+Lemma A-11

From (8), (6) we know the proposition is true under (R-≡). □