- preserves *barbs*, that is preserves some simple observational property of terms.

However, context-based behavioural equalities, such as reduction barbed congruence, suffer from the universal quantification on contexts. This quantification makes very hard to prove process equalities, and makes mechanical checking impossible. Simpler proof techniques are based on *abe ed b s ar t es* whose definitions do not use context quantification. These bisimilarities should imply, or (better) coincide with, reduction barbed congruence [24, 1, 11]. The behaviour of processes is characterised using co-inductive relations defined over a *abe ed trans t on syste* , or *L* , a collection of relations of the form

$$P \xrightarrow{\alpha} Q.$$

Intuitively the action α in the judgement $P \xrightarrow{\alpha} Q$ represents some small context with which P can interact; if the labelled bisimilarity coincides with the reduction barbed congruence then this collection of small contexts, codified as actions, is sufficient to capture all possible interactions that processes can have with arbitrary contexts.

Even if the idea of bisimulation is very general and does not rely on the specific syntax of the calculus, the definition of an appropriate notion of bisimilarity for Mobile Ambients revealed to be harder than expected. The reasons of that can be resumed as follows:

- It is difficult for an ambient n to control interferences that may originate either from other ambients in its environment or from the computation running at n itself, [17].

- Ambient mobility is asynchronous — no permission is required to migrate into an ambient. As noticed in [28], this may cause a *stutter ng* phenomenon originated by ambients that may repeatedly enter and exit another ambient. Any successful bisimilarity for MA should not observe stuttering [28].

- One of the main algebraic laws of MA is the *perfect rewa equat on*, [7]:

  $$(\nu n)n[P] = 0 \qquad \text{for } n \text{ not in } P.$$

  If you suppose $P = \mathtt{in\_}k.0$, it is evident that a bisimilarity that want to capture this law must not observe the movements of *secret a b ents*, that is those ambients, like n, whose names are not known by the rest of the system.

In [18], it is introduced a labelled bisimilarity for an “easier” variant of MA, called SAP, equipped with (i) *synchronous ob ty*, as in Levi and Sangiorgi’s *afe A b ents* [17], and (ii) *passwords* to exercise control over, and differentiate between, different ambients which may wish to exercise a capability. The main result in [18] is the characterisation of reduction barbed congruence in terms of the labelled bisimilarity. The result holds only in SAP and heavily relies on the two features (i) and (ii) mentioned above.

This work is the natural continuation of [18] where, now, we tackle the original problem: *to prov de b s u at on proof ethods for Mob e A b ents*

Contribution First of all, as in the Distributed π-calculus [14], we rewrite the syntax of MA in two levels: *processes* and *syste s*. This is because we are interested in studying systems rather than processes. So, our behavioural equalities are defined over systems. This little expedient allows us (i) to focus on higher-order actions, where movement of code is involved, and (ii) to model stuttering in terms of standard τ-actions.

We give a new labelled transition system for MA which is used to define a labelled bisimilarity over systems. The resulting bisimilarity can be defined either in *ate* or in *ear y*

Table 1 The Mobile Ambients in Two Levels

Na es		a, b, . . . , k, l, m, n, . . . □ N	
yste s			
M, N	::=	0	termination
	|	M_1 | M_2	parallel composition
	|	(νn)M	restriction
	|	n	

Table 2 Structural Congruence and Reduction Rules

$P \mid Q \equiv P \mid Q$	(Struct Par Comm)
$(P \mid Q) \mid R \equiv P \mid (Q \mid R)$	(Struct Par Assoc)
$P \mid 0 \equiv P$	(Struct Zero Par)
$(\nu n)0 \equiv 0$	(Struct Zero Res)
$!C.P \equiv C.P \mid !C.P$	(Struct Repl Par)
$(\nu n)(\nu m)P \equiv (\nu m)(\nu n)P$	(Struct Res Res)
$n \notin fn(P)$ implies $(\nu n)(P \mid Q) \equiv P \mid (\nu n)Q$	(Struct Res Par)
$n \neq m$ implies $(\nu n)(m[P]) \equiv m[(\nu n)P]$	(Struct Res Amb)

$\equiv$ is the least equivalence relation which i) satisfies the axioms and rules above and ii) is preserved by all contexts.

$n[\texttt{in\_}m.P \mid Q] \mid m[R] \rightarrow m[n[P \mid Q] \mid R]$	(Red In)
$m[n[\texttt{out\_}m.P \mid Q] \mid R] \rightarrow n[P \mid Q] \mid m[R]$	(Red Out)
$\texttt{open\_}n.P \mid n[Q] \rightarrow P \mid Q$	(Red Open)
$P \equiv Q \quad Q \rightarrow R \quad R \equiv S$ implies $P \rightarrow S$	(Red Struct)

$\rightarrow$ is the least equivalence relation which i) satisfies the rules above and ii) is preserved by all static contexts.

Definition 1.2 (Contexts) *A* **static context** *s a context where the ho e does not appear under a pre x or a rep cat on A* **system context** *s a context generated by the fo ow ng*

Table 3

Table 4 Labelled Transition System - Pre-actions

$$(\pi\ \text{Pfx})\ \frac{-}{\pi.P \xrightarrow{\pi} P} \qquad (\pi\ \text{Repl Pfx})\ \frac{-}{!\pi.P \xrightarrow{\pi} P \mid !\pi.P}$$

$$(\pi\ \text{Enter})\ \frac{P \xrightarrow{\texttt{in\_n}} P_1}{m[P] \xrightarrow{\texttt{nt r\_n}} \langle m[P_1]\rangle 0} \qquad (\pi\ \text{Amb})\ \frac{-}{n[P] \xrightarrow{\texttt{\_n}} \langle P\rangle 0}$$

$$(\pi\ \text{Exit})\ \frac{P \xrightarrow{\texttt{o t\_n}} P_1}{m[P] \xrightarrow{\texttt{t\_n}} \langle m[P_1]\rangle 0} \qquad (\pi\ \text{Res})\ \frac{P \xrightarrow{\pi} O \quad n \notin fn(\pi)}{(\nu n)P \xrightarrow{\pi} (\nu n)O}$$

$$(\pi\ \text{Par})\ \frac{P \xrightarrow{\pi} O}{\begin{array}{c} P \mid Q \xrightarrow{\pi} O \mid Q \\ Q \mid P \xrightarrow{\pi} Q \mid O \end{array}}$$

by explicitly introducing the environment's ambient interacting with the process in question. The content of this ambient will be instantiated later, in the bisimilarity, with a process. For convenience, we extend the syntax of processes with the special process $\circ$ to pinpoint those ambients whose content will be instantiated later. The process $\circ$ does not reduce: it is simply a placeholder. Notice that, unlike pre-actions and τ-actions, env-actions do not have structural rules; this is because env-actions are supposed to be performed by complete systems that can directly interact with the environment.

We call *act ons* the set of env-actions to which τ has been added. Actions always go from systems to systems and, in general, from processes to processes, even if the outcome may possibly involve the special process $\circ$. As our bisimilarity will be defined over systems, we will only consider actions (and not pre-actions) in its definition.

Proposition 2.1 *If* T *s a syste resp a process and* $T \xrightarrow{\alpha} T'$ *then* T' *s a syste resp a process poss b y conta n ng the spec a process* $\circ$

Now, we explain the rules induced by the the prefix `in`, the *grat on* of ambients. A typical example of an ambient m migrating into an ambient n is as follows:

$$(\nu m)(m[\texttt{in\_}n.P_1 \mid P_2] \mid M) \mid n[Q] \rightarrow (\nu m)(M \mid n[m[P_1 \mid P_2] \mid Q])$$

The driving force behind the migration is the activation of the prefix `in_`n, within the ambient m. It induces a capability in the ambient m to migrate into n, which we formalise as a new action `enter_`n. Thus an application of (π Enter) gives

$$m[\texttt{in\_}n.P_1 \mid P_2] \xrightarrow{\texttt{nt r\_n}} \langle m[P_1 \mid P_2]\rangle 0$$

and more generally, using the structural rules (π Res) and (π Par),

$$(\nu m)(m[\texttt{in\_}n.P_1 \mid P_2] \mid M) \xrightarrow{\texttt{nt r\_n}} (\nu m)\langle m[P_1 \mid P_2]\rangle M.$$

Table 5 Labelled Transition System - τ-actions

P

(τ Enter)

Table 6 Labelled Transition System - Env-actions

$$\textbf{(Enter)}\ \frac{\mathbf{P} \xrightarrow{\texttt{nt r\_}\mathbf{n}} (\nu\tilde{\mathbf{m}})\langle \mathbf{k}[\mathbf{P}_1]\rangle \mathbf{P}_2^{(\dagger)}}{\mathbf{P} \xrightarrow{\mathbf{k}.\ \texttt{nt r\_}\mathbf{n}} (\nu\ \tilde{}}$$

$$\mathsf{n}[\,\mathsf{m}[\mathtt{out\_n}.\mathsf{P}_1 \mid \mathsf{P}_2] \mid \mathsf{Q}] \xrightarrow{\tau} \mathsf{n}[\mathsf{Q}] \mid \mathsf{m}[\mathsf{P}_1 \mid \mathsf{P}_2].$$

Again, env-actions can model the exiting of both private and global ambients from an ambient provided by the environment.

Finally, we leave the rules which control the *open ng* as an easy exercise for the reader.

We end this section with a theorem which asserts that the LTS-based semantics coincides with the reduction semantics of Section 1.

Theorem 2.2

If $\mathsf{P} \xrightarrow{\tau} \mathsf{P}'$ *then* $\mathsf{P} \rightarrowtriangle \mathsf{P}'$

If $\mathsf{P} \rightarrowtriangle \mathsf{P}'$ *then* $\mathsf{P} \xrightarrow{\tau} \equiv \mathsf{P}'$

Proof By transition induction. Part 1 is the most di□cult. It requires a result describing the structure of a process P and the outcome O for any pre-action π

3 Characterising Reduction Barbed Congruence

In this section we define a labelled bisimilarity for MA that completely characterises reduction barbed congruence.

Since we are interested in *wea b s ar t es*, that abstract over τ-actions, we introduce the notion of weak action. The definition is standard: $\Longrightarrow$ denotes the reflexive and transitive closure of $\xrightarrow{\tau}$; $\stackrel{\alpha}{\Longrightarrow}$ denotes $\Longrightarrow \xrightarrow{\alpha} \Longrightarrow$; $\stackrel{\hat{\alpha}}{\Longrightarrow}$ denotes $\Longrightarrow$ if $\alpha = \tau$ and $\stackrel{\alpha}{\Longrightarrow}$ otherwise.

In the previous section we said that actions (and more precisely env-actions) introduce a special process $\circ$ to pinpoint those ambients whose content will be instantiated in the bisimilarity. It should be pointed out that we allow structural congruence to rearrange terms containing $\circ$: with respect to structural congruence, $\circ$ behaves like the inactive process $\mathbf{0}$. Before defining the bisimilarity we explain how $\circ$ is instantiated.

Definition 3.1 *Let* T T_1 *and* T_2 *range over both syste s and processes hen g ven a process* P *we de ne*

$$
\begin{array}{rcl@{\qquad}rcl}
\mathbf{0} \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & \mathbf{0} & (\mathsf{T}_1 \mid \mathsf{T}_2) \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & (\mathsf{T}_1 \bullet \mathsf{P}) \mid (\mathsf{T}_2 \bullet \mathsf{P}) \\
\mathsf{n}[\mathsf{R}] \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & \mathsf{n}[\mathsf{R} \bullet \mathsf{P}] & (\nu \mathsf{n})\mathsf{T} \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & (\nu \mathsf{n})(\mathsf{T} \bullet \mathsf{P}) \ \ f\, \mathsf{n} \ □ \ \mathrm{fn}(\mathsf{P}) \\
\circ \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & \mathsf{P} & \mathsf{C.R} \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & \mathsf{C.}(\mathsf{R} \bullet \mathsf{P}) \\
!\mathsf{C.R} \bullet \mathsf{P} & \stackrel{\mathrm{def}}{=} & !\mathsf{C.}(\mathsf{R} \bullet \mathsf{P}). & & &
\end{array}
$$

Now, everything is in place to define our bisimilarity.

Definition 3.2 (Late bisimilarity) *A sy etr c re at on* R *over syste s s a* **late bisimulation** *f* M R N *p es*

- *f* $\mathsf{M} \xrightarrow{\alpha} \mathsf{M}'$ α □ {□.enter_n, □.exit_n} *then there s a syste* N' *such that* $\mathsf{N} \stackrel{\hat{\alpha}}{\Longrightarrow} \mathsf{N}'$ *and for a processes* P *t ho ds* $\mathsf{M}' \bullet \mathsf{P}$ R $\mathsf{N}' \bullet \mathsf{P}$
- *f* $\mathsf{M} \xrightarrow{□.\,\mathtt{nt\ r\_n}} \mathsf{M}'$ *then there s a syste* N' *such that* $\mathsf{N} \mid \mathsf{n}[\circ] \Longrightarrow \mathsf{N}'$ *and for a processes* P *t ho ds* $\mathsf{M}' \bullet \mathsf{P}$ R $\mathsf{N}' \bullet \mathsf{P}$
- *f* $\mathsf{M} \xrightarrow{□.\,\mathtt{t\_n}} \mathsf{M}'$ *then there s a syste* N' *such that* $\mathsf{n}[\circ \mid \mathsf{N}] \Longrightarrow \mathsf{N}'$ *and for a processes* P *t ho ds* $\mathsf{M}' \bullet \mathsf{P}$ R $\mathsf{N}' \bullet \mathsf{P}$

M *and* N *are* **late bisimilar** *wr tten* $\mathsf{M} \approx \mathsf{N}$ *f* M R N *for so e ate b s u at on* R

The bisimilarity above has a universal quantification over the process P provided by the environment. This process instantiates the special process $\circ$ generated via env-actions. The bisimilarity is defined in a *ate* style as the existential quantification precedes the universal one. Another possibility would be to define the bisimilarity in *ear y* style where the universal quantification over the environment's contribution P precedes that over the derivative N'. We write $\approx_e$ to denote this early variant. By definition, every late bisimulation is also a early one, while the converse, in general, does not hold. However, in our case, as in HOπ [25], we will prove that late and early bisimilarity actually coincide. As a consequence, late bisimilarity will be our main labelled bisimilarity because the derivatives N' do not depend on processes P.

Finally, notice that, in the definition of bisimilarity, actions $\square.\mathtt{enter}\_n$ and $\square.\mathtt{exit}\_n$ are treated apart asking for weaker matching requirements. This is because both actions are not observable. Somehow, this is very similar to what happens with input actions in the asynchronous π-calculus [15, 3].

3.1 Soundness

Late and early bisimilarity represent two proof techniques for reduction barbed congruence. More precisely we prove that they are both contextual and contained in reduction barbed congruence.

The following lemma is crucial for proving that $\approx$ is contextual. This lemma will be also used for proving the soundness of the up-to-context proof techniques in Section 4.

Lemma 3.3 *Let* S *be a contextua sy etr c re at on between syste s Let* $(M, N) \in S$ *be a pa r sat sfy ng the b s u at on cond t ons n* S *that s*

f $M \xrightarrow{\alpha} M'$ $\alpha \notin \{\square.\mathtt{enter}\_n, \square.\mathtt{exit}\_n\}$ *then there s a syste* N' *such that* $N \stackrel{\hat{\alpha}}{\Longrightarrow} N'$ *and for a processes* P *t ho ds* $M' \bullet P \; S \; N' \bullet P$

f $M \xrightarrow{\square.\mathtt{enter}\_n} M'$ *then there s a syste* N' *such that* $N \mid n[\circ] \Longrightarrow N'$ *and for a processes* P *t ho ds* $M' \bullet P \; S \; N' \bullet P$

f $M \xrightarrow{\square.\mathtt{exit}\_n} M'$ *then there s a syste* N' *such that* $n[\circ \mid N] \Longrightarrow N'$ *and for a processes* P *t ho ds* $M' \bullet P \; S \; N' \bullet P$

hen a the pa rs $(C[M], C[N])$ *for any syste context* $C[-]$ *a so sat sfy the b s u at on cond t ons n* S

Proof The relation S is contextual, and as such it is the smallest relation between systems such that:

- if $M \; S \; N$, then $M \mid H \; S \; N \mid H$ for all systems H;
- if $M \; S \; N$, then $(\nu m)M \; S \; (\nu m)N$ for all names m;
- if $M \; S \; N$, then $m[M \mid P] \; S \; m[N \mid P]$ for all names m and processes P.

We prove the closure of $C[M] \; S \; C[N]$ under the conditions for being a bisimulation by induction on the structure of $C[-]$.

- $C[-] = -$.

  This case holds because $M \; S \; N$

– $(\nu m)D[M] \xrightarrow{\tau} O_1$.

This can only be derived from $D[M] \xrightarrow{\tau} O_1$, where $O_1 = (\nu m)O_1$. The induction hypothesis tells us that there exists a system O_2 such that $D[N] \Rrightarrow O_2$ and $O_1 \; S \; O_2$. We can derive $(\nu m)D[N] \Rrightarrow (\nu m)O_2$ and conclude $(\nu m)O_1 \; S \; (\nu m)O_2$ because S is closed under restriction.

– $(\nu m)D[M] \xrightarrow{k.\;nt\;r\_n} O_1$.

Observe that this must have been derived from

$$\dfrac{\dfrac{D[M] \xrightarrow{nt\;r\_n} (\nu \tilde{r})\langle k[M_1]\rangle M_2}{(\nu m)D[M] \xrightarrow{nt\;r\_n} (\nu m)(\nu \tilde{r})\langle k[M_1]\rangle M_2}}{(\nu m)D[M] \xrightarrow{k.\;nt\;r\_n} O_1 \;\equiv\; (\nu m)(\nu \tilde{r})(n[\circ \mid k[M_1]] \mid M_2)}$$

for some process M_1 and system M_2. Remark that this implies $m \boxminus n$ and $m \boxminus k$. Ask.

$(\nu m)N' = O_2$. We can conclude that for all processes P, it holds $O_1 \bullet P \; S \; O_2 \bullet P$ up to structural congruence, because S is closed under restriction.

- $(\nu m)D[M] \xrightarrow{n.\overline{\mathtt{nt\ r}}\_k} O_1$.
  Observe that this must have been derived from

$$D[M] \xrightarrow{\_n} (\nu \tilde{r})\langle$$

– $(\nu m)D[M] \xrightarrow{\square.\ nt\ r\_n} O_1$.

Observe that there are two possible derivations.

 Suppose:

$$\frac{\dfrac{D[M] \xrightarrow{nt\ r\_n} (\nu \tilde{r})\langle m[M_1]\rangle M_2}{(\nu m)D[M] \xrightarrow{nt\ r\_n} (\nu m)(\nu \tilde{r})\langle m[M_1]\rangle M_2}}{(\nu m)D[M] \xrightarrow{\square.\ nt\ r\_n} O_1 \equiv (\nu m)(\nu \tilde{r})(n[\circ \mid m[M_1]] \mid M_2)}$$

where m

where m □ r̃, for some process M_1 and system M_2. Remark that this implies n □ r. As

□ Suppose:

$$\frac{\dfrac{D[M] \xrightarrow{-:\,t\_n} (\nu\tilde{r})\langle k[M_1]\rangle M_2}{D[M] \mid H \xrightarrow{-:\,t\_n} (\nu\tilde{r})\langle k[M_1]\rangle M_2 \mid H}}{D[M] \mid H \xrightarrow{k.\,-:\,t\_n} O_1 \;\equiv\; (\nu\tilde{r})(n[\circ \mid M_2 \mid H] \mid k[M_1])}$$

for some process M_1 and system M_2. Remark that k □ $\tilde{r}$. As $D[M] \xrightarrow{-:\,t\_n} (\nu\tilde{r})\langle k[M_1]\rangle M_2$ then $D[M] \xrightarrow{k.\,-:\,t\_n} (\nu\tilde{r})(n[\circ \mid M_2] \mid k[M_1]) = M'$. The induction hypothesis then tells us that there exist systems N', A, B such that $D[N] \Rightarrow A \xrightarrow{k.\,-:\,t\_n} B \Rightarrow N'$, and for all processes P it holds $M' \bullet P \; S \; N' \bullet P$. Remark that $N' \equiv (\nu\tilde{h})n[\circ \mid N_3] \mid N_4$, for some N_3, N_4. As $A \xrightarrow{k.\,-:\,t\_n} B$, the system B must be of the form $(\nu\tilde{s})(n[\circ \mid N_2] \mid k[N_1])$, for some process N_1 and system N_2. It also holds $A \xrightarrow{-:\,t\_n} (\nu\tilde{s})\langle k[N_1]\rangle N_2$. This implies $A \mid H \xrightarrow{-:\,t\_n} (\nu\tilde{s})\langle k[N_1]\rangle N_2 \mid H$, from which we can derive $A \mid H \xrightarrow{k.\,-:\,t\_n} (\nu\tilde{s})(n[\circ \mid N_2 \mid H] \mid k[N_1]) \equiv B \bullet (\circ \mid H)$. We obtain $D[N] \mid H \Rightarrow A \mid H \xrightarrow{k.\,-:\,t\_n} B \bullet (\circ \mid H) \Rightarrow\equiv N' \bullet (\circ \mid H)$. Call $N' \bullet (\circ \mid H) = O_2$. As for all processes P it holds $M' \bullet P \; S \; N' \bullet P$, we can conclude that for all processes Q, it holds $O_1 \bullet Q \; S \; O_2 \bullet Q$ up to structural congruence, because $O_1 \bullet Q \equiv M' \bullet (Q \mid H) \; S \; N' \bullet (Q \mid H) \equiv O_2 \bullet Q$.

□ Suppose:

$$\frac{\dfrac{H \xrightarrow{-:\,t\_n} (\nu\tilde{r})\langle k[H_1]\rangle H_2}{D[M] \mid H \xrightarrow{-:\,t\_n} (\nu\tilde{r})\langle k[H_1]\rangle H_2 \mid D[M]}}{D[M] \mid H \xrightarrow{k.\,-:\,t\_n} O_1 \;\equiv\; (\nu\tilde{r})(n[\circ \mid H_2 \mid D[M]] \mid k[H_1])}$$

for some process H_1 and system H_2. Remark that k □ $\tilde{r}$. We can construct the following 0x02 3.24883 0 Td [(M)0.228719]TJ /R10 11.9551 Tf 12.573 0 Td [(])0.216 S])

- Suppose:

$$\dfrac{\dfrac{D[M] \xrightarrow{\bar{n}} (\nu \tilde{r})\langle M_1 \rangle M_2}{D[M] \mid H \xrightarrow{\bar{n}} (\nu \tilde{r})\langle M_1 \rangle M_2 \mid H}}{D[M] \mid H \xrightarrow{n.\overline{nt\ r}\_k} O_1 \equiv (\nu \tilde{r})(n[k[\circ] \mid M_1] \mid M_2 \mid H)}$$

for some process M_1 and system M_2. Remark that k, n □ $\tilde{r}$. As $D[M] \xrightarrow{\bar{n}} (\nu \tilde{r})\langle M_1 \rangle M_2$ then $D[M] \xrightarrow{n.\overline{nt\ r}\_k} (\nu \tilde{r})(n[k[\circ] \mid M_1] \mid M_2) = M'$. The induction hypothesis then tells us that there exist systems N', A, B such that $D[N] \Rrightarrow A \xrightarrow{n.\overline{nt\ r}\_k} B \Rrightarrow N'$, and for all processes P it holds $M' \bullet P \; S \; N' \bullet P$. As $A \xrightarrow{n.\overline{nt\ r}\_k} B$, the system B must be of the form $(\nu \tilde{s})(n[k[\circ] \mid N_1] \mid N_2)$, for some process N_1 and system N_2. It also holds $A \xrightarrow{\bar{n}} (\nu \tilde{s})\langle N_1 \rangle N_2$. This implies $A \mid H \xrightarrow{\bar{n}} (\nu \tilde{s})\langle N_1 \rangle N_2 \mid H$, from which we can derive $A \mid H \xrightarrow{n.\overline{nt\ r}\_k} (\nu \tilde{s})(n[k[\circ] \mid N_1] \mid N_2 \mid H) \equiv B \mid H$. We obtain $D[N] \mid H \Rrightarrow A \mid H \xrightarrow{n.\overline{nt\ r}\_k} \equiv B \mid H \Rrightarrow \equiv N' \mid H$. Call $N' \mid H = O_2$. We can conclude that for all processes P, it holds $O_1 \bullet P \; S \; O_2 \bullet P$ up to structural congruence, because S is closed under parallel composition.

- Suppose:

$$\dfrac{\dfrac{H \xrightarrow{\bar{n}} (\nu \tilde{r})\langle H_1 \rangle H_2}{D[M] \mid H \xrightarrow{\bar{n}} (\nu \tilde{r})\langle H_1 \rangle H_2 \mid D[M]}}{D[M] \mid H \xrightarrow{n.\overline{nt\ r}\_k} O_1 \equiv (\nu \tilde{r})(n[k[\circ] \mid H_1] \mid H_2 \mid D[M])}$$

for some process H_1 and system H_2. Remark that k □ $\tilde{r}$. We can construct the following derivation:

$$\dfrac{\dfrac{H \xrightarrow{\bar{n}} (\nu \tilde{r})\langle H_1 \rangle H_2}{D[N] \mid H \xrightarrow{\bar{n}} (\nu \tilde{r})\langle H_1 \rangle H_2 \mid D[N]}}{D[N] \mid H \xrightarrow{n.\overline{nt\ r}\_k} (\nu \tilde{r})(n[k[\circ] \mid H_1] \mid H_2 \mid D[N]) = O_2}$$

We can conclude that for all processes P, it holds $O_1 \bullet P \; S \; O_2 \bullet P$ up to structural congruence, because $D[M] \; S \; D[N]$ and S is clo R69 7.97011 Tf ..97011 Tf ..

 Suppose:

$$\cfrac{\cfrac{\mathbf{D[M]} \xrightarrow{\bar{\ }\mathbf{n}} (\nu\tilde{\mathbf{r}})\langle\mathbf{M}_1\rangle\mathbf{M}_2}{\mathbf{D[M]} \mid \mathbf{H} \xrightarrow{\bar{\ }\mathbf{n}} (\nu\tilde{\mathbf{r}})\langle\mathbf{M}_1\rangle\mathbf{M}_2 \mid \mathbf{H}}}{\mathbf{D[M]} \mid \mathbf{H} \xrightarrow{\mathbf{k}.\mathrm{op}\ \mathrm{n}\_\mathbf{n}} \mathbf{O}_1 \equiv \mathbf{k}[\circ \mid (\nu\tilde{\mathbf{r}})(\mathbf{M}_1 \mid \mathbf{M}_2) \mid \mathbf{H}]}$$

for some process $\mathbf{M}_1$ and system $\mathbf{M}_2$. Remark that k, n r̃. As D[M] $\xrightarrow{\bar{\ }\mathbf{n}}$ (ν r̃)

Suppose:

$$\mathbf{D[M]} \xrightarrow{\texttt{nt r\_}\mathbf{n}} (\nu \tilde{\mathbf{r}})$$

$n[\circ \mid D[N]] \;\Rightarrow\; N'$, and for all processes P it holds $M' \bullet P \; S \; N' \bullet P$. Remark that $N' \equiv (\nu \tilde{s})n[\circ \mid N_3] \mid N_4$, for some N_3, N_4. We can derive $n[\circ D[N] \mid H] \;\Rightarrow\; (\nu\tilde{s})n[\circ \mid N_3 \mid H] \mid N_4$. Call $(\nu\tilde{s})n[\circ \mid N_3 \mid H] \mid N_4 = O_2$. As for all processes P it holds $M' \bullet P \; S \; N' \bullet P$, we can conclude that for all processes Q, it holds $O_1 \bullet Q \; S \; O_2 \bullet Q$ up to structural congruence, because $O_1 \bullet Q \equiv M' \bullet (Q \mid H) \; S \; N' \bullet (Q \mid H) \equiv O_2 \bullet Q$.

- Suppose:

$$\dfrac{\dfrac{H \xrightarrow{\text{-: t\_n}} (\nu\tilde{r})\langle k[H_1]\rangle H_2}{D[M] \mid H \xrightarrow{\text{-: t\_n}} (\nu\tilde{r})\langle k[H_1]\rangle H_2 \mid D[M]}}{D[M] \mid H \xrightarrow{\square.\text{-: t\_n}} O_1 \;\equiv\; (\nu\tilde{r})(n[\circ \mid H_2 \mid D[M]] \mid k[H_1])}$$

for some process H_1 and system H_2. Remark that k □ $\tilde{r}$. We can construct the following derivation:

$$\dfrac{\dfrac{H \xrightarrow{\text{-: t\_n}} (\nu\tilde{r})\langle k[H_1]\rangle H_2}{D[N] \mid H \xrightarrow{\text{-: t\_n}} (\nu\tilde{r})\langle k[H_1]\rangle H_2 \mid D[N]}}{n[\circ \mid D[N] \mid H] \xrightarrow{\tau} (\nu\tilde{r})(n[\circ \mid H_2 \mid D[N]] \mid k[H_1]) = O_2}$$

We can conclude that for all processes P, it holds $O_1 \bullet P \; S \; O_2 \bullet P$ up to structural congruence, because $D[M] \; S \; D[N]$ and S is closed under parallel composition and ambient.

Then, we consider the cases when there is interaction between $D[M]$ and H.

– $D[M] \mid H \xrightarrow{\tau} O_1$, because

$$D[M] \xrightarrow{\text{nt r\_n}} (\nu\tilde{m})\langle k[M_1]\rangle M_2 \text{ and } H \xrightarrow{\text{\_n}} (\nu\tilde{h})\langle H_1\rangle H_2.$$

Then $O_1 \equiv (\nu\tilde{h},\tilde{m})(n[k[M_1] \mid H_1] \mid M_2 \mid H_2)$. We distinguish the cases k □ $\tilde{m}$, and k □ $\tilde{m}$.

- k □ $\tilde{m}$. As $D[M] \xrightarrow{\text{nt r\_n}} (\nu\tilde{m})\langle k[M_1]\rangle M_2$, it also holds $D[M] \xrightarrow{\text{k. nt r\_n}} M' \equiv (\nu\tilde{m})(n[\circ \mid k[M_1]] \mid M_2)$. The induction hypothesisn constr

H_1

□M

exists a system N' such that $D[N] \mid n[\circ] \Rrightarrow N' \equiv (\nu \tilde{n})(n[\circ \mid N_1] \mid N_2)$, and

□ k □ r̃

Theorem 3.4 *Late b s ar ty s contextua*

Proof Let S be the smallest binary relation between systems such that:

1. $\approx$ □ S;
2. if $M\ S\ N$, then $C[M]\ S\ C[N]$ for all system contexts $C[-]$.

Remark that S is symmetric because of the symmetry of $\approx$. We prove that S is a late bisimilarity up to $\equiv$ by induction on the definition of S.

- $M\ S\ N$ because $M \approx N$.

  Immediate.

- $C[M]\ S\ C[N]$ because $M\ S\ N$.

  The induction hypothesis assures that (M, N) □ S is a pair satisfying the bisimulation conditions in S. Lemma 3.3 assures that the pair $(C[M], C[N])$ satisfies the bisimulation conditions in S.

□

Note that the above proof does not rely on the transitivity of the late bisimulation. Note also that it is easy to adapt Lemma 3.3 and the above proof to show that early bisimilarity is contextual.

Proposition 3.5 *Late b s ar ty s an equ va ence re at on*

Proof

Table 7 Contexts for visible actions

$\alpha = k.\mathtt{enter}\_n$	$C_\alpha[-] = n[\circ \mid \ \mathtt{on}\bullet[\mathtt{in}\_k.\mathtt{out}\_k.\mathtt{out}\_n]] \mid -$
$\alpha = k.\mathtt{exit}\_n$	$C_\alpha[-] = (\nu a)a[\mathtt{in}\_k.\mathtt{out}\_k.\ \mathtt{on}\bullet[\mathtt{out}\_a]] \mid n[\circ \mid -]$
$\alpha = n.\overline{\mathtt{enter}}\_k$	$C_\alpha[-] = (\nu a)a[\mathtt{in}\_n.k[\mathtt{out}\_a.(\circ \mid (\nu b)b[\mathtt{out}\_k.\mathtt{out}\_n.\ \mathtt{on}\bullet[\mathtt{out}\_b]])]] \mid -$
$\alpha = k.\mathtt{open}\_n$	$C_\alpha[-] = k[\circ \mid (\nu a, b)(\mathtt{open}\_b.\mathtt{open}\_a.\ \mathtt{on}\bullet[\mathtt{out}\_k] \mid a[- \mid \mathtt{open}\_n.b[\mathtt{out}\_a]]))]$
	where a and b are re r17034]TJ R227.343(8.98398 0 Td T.9551 Tf 22.21 4.85938 re f

This implies $C_{k.\mathtt{nt\ r\_}n}[M] \bullet P \Longrightarrow M' \bullet P \mid$ on $[\,]$.

$\alpha = k.\mathtt{exit\_}n$ Let P be a process. We know that $M \xrightarrow{k.\,\cdot\,\mathtt{t\_}n} M'$

Table 8 Auxiliary contexts and processes

$$-_1 \,\square\, -_2 \quad = \quad (\nu o)(o[\,]o$$

Proof

there exist systems $\mathbf{M}_1$ and $\mathbf{M}_2$ and a static context $\mathbf{C}[-]$ such that:

$$
\begin{aligned}
& \mathbf{C}_{\alpha}[\mathbf{M}] \bullet \mathbf{SPY}_{\alpha}\langle \mathbf{i}, \mathbf{j}, \mathbf{P} \rangle \\
= \quad & \mathbf{n}[\mathbf{SPY}_{\alpha}\langle \mathbf{i}, \mathbf{j}, \mathbf{P} \rangle \mid \mathtt{om}\bullet[\mathtt{in\_}\mathbf{k}.\mathtt{out\_}\mathbf{k}.\mathtt{out\_}\mathbf{n}]] \mid \mathbf{M}
\end{aligned}
$$

As the name on● is fresh for M, by several applications of Lemma 3.11 to the reduction marked by (□) we have:

$$(\nu a)a[\mathtt{in\_}k.\mathtt{out\_}k.0] \mid M_1 \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle$$
$$\Rrightarrow (\nu a)E[0 \mid a[\,]] \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle.$$

Again, as a is fresh, by several applications of Lemma 3.11, and reducing underneath (νa), we obtain:

$$(\nu a)(0 \mid M_1) \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle$$
$$\Rrightarrow (\nu a)E[0 \mid 0] \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle.$$

Summarising,

$$M_1 \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle \equiv (\nu a)(0 \mid M_1) \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle \;\Rrightarrow (\nu a)E[0 \mid 0] \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle$$

and, as $\equiv$ is closed under reductions,

$$M_1 \;\Rrightarrow\equiv E[0].$$

So, assuming $M' = E[0]$, we can conclude.

$\alpha = n.\overline{\mathtt{enter}}\_k$. Observe that

$$C_\alpha[M] \bullet \mathrm{SPY}_\alpha\langle i, j, P\rangle =$$
$$(\nu a)a[\mathtt{in\_}n.k[\mathtt{out\_}a.(\mathrm{SPY}_\alpha\langle i, j, P\rangle \mid (\nu b)b[\mathtt{out\_}k.\mathtt{out\_}n.\ \mathtt{on}\bullet[\mathtt{out\_}b]])]] \mid M$$

To unleash the ambient on●, the ambient a must use its in_n capability, and the aifbitEthed2t5.49apathi thiiimplie thaatheoa4.236.896(t)-0.147034(h)-0.31047032 reductiocstsecrea

Observe that,

$$D[(\nu a)a[k[\mathtt{out\_}a.(SPY_{\alpha}\langle i, j, P\rangle \mid (\nu b)b[\mathtt{out\_}k.\mathtt{out\_}n.\ \mathtt{on}\bullet[\mathtt{out\_}b]]])]]$$
$$\sqsubseteq D[k[SPY_{\alpha}\langle i, j, P\rangle \mid (\nu b)b[\mathtt{out\_}k.\mathtt{out\_}n.\ \mathtt{on}\bullet[\mathtt{out\_}b]]]]$$

Thus, by examining the above reductions sequence from $C_{n.\overline{\mathtt{nt\ r}}\_k}[$

ambient a. More precisely, there exist a system M_1, processes Q_i, and a static context $D[-]$ such that:

$$
\begin{array}{ll}
 & C_{k.op\_n\_n}[M] \bullet SPY_\alpha\langle i,j,P\rangle \\
= & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid a[M \mid \mathtt{open\_}n.b[\mathtt{out\_}a]])] \\
\Rrightarrow & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid a[M_1 \mid \mathtt{open\_}n.b[\mathtt{out\_}a]])] \\
\xrightarrow{\tau} & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid a[Q \mid b[\mathtt{out\_}a]])] \\
\Rrightarrow & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid a[Q_1 \mid b[\mathtt{out\_}a]])] \\
\xrightarrow{\tau} & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid b[\,] \mid a[Q_1])] \\
\Rrightarrow & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid b[\,] \mid a[Q_2])] \\
\xrightarrow{\tau} & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid 0 \mid a[Q_2])] \\
\Rrightarrow & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid 0 \mid a[Q_3])] \\
\Rrightarrow & k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid 0 \mid Q_3)] \\
\Rrightarrow & D[\ \mathtt{on}\bullet[\,]] \bullet SPY_\alpha\langle i,j,P\rangle \\
\equiv & D[0] \bullet SPY_\alpha\langle i,j,P\rangle \mid\ \mathtt{on}\bullet[\,] \\
= & O \mid\ \mathtt{on}\bullet[\,]
\end{array}
$$

Examining the above reductions sequence from $C_{k.op\_n\_n}[M] \bullet SPY_\alpha\langle i,j,P\rangle$ we conclude that

$$M \Rrightarrow \xrightarrow{k.op\_n\_n} k[\circ \mid Q].$$

As

$$
\begin{array}{l}
k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.\ \mathtt{on}\bullet[\mathtt{out\_}k] \mid a[Q \mid b[\mathtt{out\_}a]])] \\
\quad \Rrightarrow D[\ \mathtt{on}\bullet[\,]] \bullet SPY_\alpha\langle i,j,P\rangle
\end{array}
$$

and the name $\mathtt{on}\bullet$ is fresh, by several applications of Lemma 3.11 we have

$$
\begin{array}{l}
k[SPY_\alpha\langle i,j,P\rangle \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.0 \mid a[Q \mid b[\mathtt{out\_}a]])] \\
\quad \Rrightarrow D[0] \bullet SPY_\alpha\langle i,j,P\rangle.
\end{array}
$$

By Lemma 3.10, this implies

$$k[\circ \mid (\nu a,b)(\mathtt{open\_}b.\mathtt{open\_}a.0 \mid a[Q \mid b[\mathtt{out\_}a]])] \Rrightarrow D[0].$$

Applying our proof techniques we can easily prove that:

$$k[\circ \mid (\nu a,b)(\mathtt{open\_}$$

$\Rightarrow$ $k.op\ n$

Theorem 3.13 (Completeness)

To conclude we must establish that for all P, it holds $M' \bullet P \cong N' \bullet P$. As barbed congruence is preserved by restriction, we have (ν

where $M' \bullet SPY_{*.\ \ t}n(i, j, P)\ _{i,J}$. Call this outcome O_1.

This reduction must be matched by a corresponding reduction

$$C(P)[N] \Rightarrow O_2$$

where $O_1 \sqsubseteq O_2$ and $O_2\ _{A,B}$. By several applications of Lemma 3.10 it follows that there is a system N' such that $O_2 = N' \bullet SPY_{*.\ nt\ r}n(i, j, P$

- f $\mathbf{M} \xrightarrow{\alpha} \mathbf{M}''$ α □ $\{\texttt{enter\_}\mathbf{n}, \texttt{exit\_}\mathbf{n}\}$ *then there ex sts a syste* $\mathbf{N}''$ *such that* $\mathbf{N} \stackrel{\hat{\alpha}}{\Longrightarrow} \mathbf{N}''$ *and for a processes* $\mathbf{P}$ *there s a syste context* $\mathbf{C}[-]$ *and syste s* $\mathbf{M}'$ *and* $\mathbf{N}'$ *such that* $\mathbf{M}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{M}']$ $\mathbf{N}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{N}']$ *and* $\mathbf{M}' \; \mathbf{R} \; \mathbf{N}'$

- f $\mathbf{M} \xrightarrow{\texttt{nt r\_}\mathbf{n}} \mathbf{M}''$ *then there ex sts a syste* $\mathbf{N}''$ *such that* $\mathbf{N} \mid \mathbf{n}[\circ] \Longrightarrow \mathbf{N}''$ *and for a processes* $\mathbf{P}$ *there s a syste context* $\mathbf{C}[-]$ *and syste s* $\mathbf{M}'$ *and* $\mathbf{N}'$ *such that* $\mathbf{M}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{M}']$ $\mathbf{N}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{N}']$ *and* $\mathbf{M}' \; \mathbf{R} \; \mathbf{N}'$

- f $\mathbf{M} \xrightarrow{\texttt{-: t\_}\mathbf{n}} \mathbf{M}''$ *then there ex st a syste* $\mathbf{N}''$ *such that* $\mathbf{n}[\circ \mid \mathbf{N}] \Longrightarrow \mathbf{N}''$ *and for a processes* $\mathbf{P}$ *there s a syste context* $\mathbf{C}[-]$ *and syste s* $\mathbf{M}'$ *and* $\mathbf{N}'$ *such that* $\mathbf{M}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{M}']$ $\mathbf{N}'' \bullet \mathbf{P} \gtrsim \mathbf{C}[\mathbf{N}']$ *and* $\mathbf{M}' \; \mathbf{R} \; \mathbf{N}'$

Theorem 4.7 *If* $\mathbf{R}$ *s a b s u at on up to context and up to* $\gtrsim$ *then* $\mathbf{R}$ □ $\approx$

Proof We define the relation S as the smallest relation such that:

1. $\mathbf{M} \; \mathbf{R} \; \mathbf{N}$ implies $\mathbf{M} \; \mathbf{S} \; \mathbf{N}$;

2. $\mathbf{M} \gtrsim \mathbf{A}, \mathbf{A} \; \mathbf{S} \; \mathbf{B}, \mathbf{B} \lesssim \mathbf{N}$ implies $\mathbf{M} \; \mathbf{S} \; \mathbf{N}$;

3. $\mathbf{M} \; \mathbf{S} \; \mathbf{N}$ implies $\mathbf{C}[\mathbf{M}] \; \mathbf{S} \; \mathbf{C}[\mathbf{N}]$, for all system contexts $\mathbf{C}[-]$.

We prove by induction on its definition, that S is a late bisimulation. This will assure the soundness of the relation R, because $\mathbf{M} \; \mathbf{R} \; \mathbf{N}$ implies $\mathbf{M} \; \mathbf{S} \; \mathbf{N}$ which implies $\mathbf{M} \approx \mathbf{N}$

- $M \; S \; N$ because $M \gtrsim A, A \; S \; B, B \lesssim N$.

  The induction hypothesis tells us that $A \; S \; B$ behaves like a late bisimulation.

  Suppose $M \xrightarrow{\alpha} M'$, with $\alpha \notin \{\square.\mathtt{enter}\_n, \square.\mathtt{exit}\_n\}$. A simple diagram chasing allows us to conclude that there are systems A', B', N' such that for all process P it holds $M' \bullet P \gtrsim A' \bullet P \; S \; B' \bullet P \lesssim N' \bullet P$, and in turn, by construction of S, $M' \bullet P \; S \; N' \bullet P$.

  Suppose $M \xrightarrow{\square.\mathtt{enter}\_n} M'$. As $M \gtrsim A$, for all process P, it holds $M' \bullet P \gtrsim A \mid n[P]$. As $A \; S \; B$, the closure properties of S assure that $A \mid n[P] \; S \; B \mid n[P]$. The expansion relation is a congruence, and since $B \; S \; N$ we conclude that $B \mid n[P] \lesssim N \mid n[P]$. But $N \mid n[P] \Rrightarrow N \mid n[P]$, and $M' \bullet P \gtrsim S \lesssim (N \mid n[\circ]) \bullet P$. This, by construction of S, implies $M' \bullet P \; S \; (N \mid n[\circ]) \bullet P$.

  Suppose $M \xrightarrow{\square.\mathtt{exit}\_n} M'$. As $M \gtrsim A$, for all process P, it holds $M' \bullet P \gtrsim n[P \mid A]$. As $A \; S \; B$, the closure properties of S assure that $n[P \mid A] \; S \; n[P \mid B]$. The expansion relation is a congruence, and since $B \; S \; N$ we conclude that $n[P \mid A] \lesssim n[P \mid N]$. But $n[P \mid B] \Rrightarrow n[P \mid N]$, and $M' \bullet P \gtrsim S \lesssim n[\circ \mid N] \bullet P$. This, by construction of S, implies $M' \bullet P \; S \; n[\circ \mid N] \bullet P$.

- $C[M] \; S \; C[N]$ because $M \; S \; N$ and $C[-]$ is a system context.

  The induction hypothesis tells us that $(M, N) \in S$ is a pair satisfying the bisimulation conditions in S. Lemma 3.3 assures that the pair $(C[M], C[N]) \in S$ satisfies the bisimulation conditions in S.

$\square$

5 Adding Communication

The basic idea is to have an *output process* such as $\langle E \rangle.P$, which outputs the message E and then continues as P, and an input process $(x)Q$ which on receiving a message binds it to x in Q which then executes; here occurrences of x in Q are bound. Notice that we have synchronous output; as discussed in [illegible]

Table 9 The Message-passing Mobile Ambients in Two Levels

Names		$a, b, \ldots, k, l, m, n, \ldots \in N$	
Capabilities			
C	$::=$	$\mathtt{in}\_n$	may enter into n
	$\mid$	$\mathtt{out}\_n$	may exit out of n
	$\mid$	$\mathtt{open}\_n$	may open n
Expressions			
E, F	$::=$	x	variable
	$\mid$	C	capability
	$\mid$	$E.F$	path
	$\mid$	ε	empty path
Guards			
G	$::=$	E	expression
	$\mid$	(x)	input
	$\mid$	$\langle E \rangle$	output
Systems			
M, N	$::=$	0	termination
	$\mid$	$M_1 \mid M_2$	parallel composition
	$\mid$	$(\nu n)M$	restriction
	$\mid$	$n[P]$	ambient
Processes			
P, Q, R	$::=$	0	nil process
	$\mid$	$P_1 \mid P_2$	parallel composition
	$\mid$	$(\nu n)P$	restriction
	$\mid$	$G.P$	prefixing
	$\mid$	$n[P]$	ambient
	$\mid$	$!G.P$	replication

Structural and Reduction rules for Communication

$E.(F.P) \equiv (E.F).P$	(Struct Path)
$\varepsilon.P \rightarrow P$	(Red Empty Path)
$(x).P \mid \langle M \rangle.Q \rightarrow P\{^M/_x\} \mid Q$	(Red Comm)

Table 10 Pre-actions and Concretions for Communication

Pre-actions	π	$::= \ldots \mid (E) \mid \langle - \rangle$	*Concretions*	K	$::=$	$(\nu \tilde{m})\langle P \rangle Q$
					$\mid$	$(\nu \tilde{m})\langle E \rangle Q$

Table 11 Labelled Transition System - Communication

$$(\pi\ \text{Output})\ \frac{-}{□E□.P \xrightarrow{□-□} \langle E \rangle P} \qquad (\pi\ \text{Input})\ \frac{-}{(x).P \xrightarrow{(E)} P\{^{E}/_{x}\}}$$

$$(\pi\ \text{Path})\ \frac{E.(F.P) \xrightarrow{\pi} Q}{(E.F).P \xrightarrow{\pi} Q} \qquad (\tau\ \text{Eps})\ \frac{-}{□.P \xrightarrow{\tau} P}$$

$$(\tau\ \text{Comm})\ \frac{P \xrightarrow{□-□} (\nu\tilde{m})\langle E \rangle P' \quad Q \xrightarrow{(E)} Q' \quad fn(Q') \cap \{\tilde{m}\} = □}{P \mid Q \xrightarrow{\tau} (\nu\tilde{m})(P' \mid Q')}$$

instantiated by a system context, because in a system context the hole cannot appear under a prefix. This in turn implies that our bisimulations can be applied to the extended calculus, and all the results of Section 3 and Section 4 hold without modifications.

Theorem 5.1 *Late b s ar ty ear y b s ar ty and barbed congruence co nc de n the Message Pass ng Ca cu us*

Theorem 5.2 *he up to expans on up to context and up to context and expans on proof techn ques are sound proof techn ques n the Message Pass ng Ca cu us*

6 Algebraic Theory

In this section we prove a a collection of algebraic propertie

$(\nu n)n[m[\mathtt{out\_}n.P] \mid \prod_{j\in J} \mathtt{open\_}k_j.R_j] \;\square\; (\nu n)(m[P] \mid n[\prod_{j\in J} \mathtt{open\_}k_j.R_j])$ f m □ k_j *for* j □ J

$n[(\nu m)(\mathtt{open\_}m.P \mid m[N]) \mid Q] \;\square\; n[(\nu m)(P \mid N) \mid Q]$ f $Q \equiv M \mid \prod_{j\in J}$ □W_j□$.R_j$ *and*

Lemma 6.2 *Let* **P** **Q** *and* **R** *be processes* *hen*

$$(\nu k, m, w)(k[\mathtt{in\_}m.P] \mid m[\mathtt{open\_}k.Q] \mid w[\mathtt{open\_}m.R])$$
$$\cong (\nu k, m, w)(m[k[P] \mid \mathtt{open\_}k.Q] \mid w[\mathtt{open\_}m.R])$$

$$(\nu m, w)(m[\langle \mathtt{in\_}w \rangle \mid (x).P] \mid w[\mathtt{open\_}m.Q]) \cong (\nu m, w)(m[P\{^{\mathtt{in\_}w}/_{x}\}] \mid w[\mathtt{open\_}m.Q])$$

Proof

- **Our env-actions, unlike those in [18], are truly late, as they do not mention the process provided by the environment. This process can be added** *ate*, **when playing the bisimulation game.**

- **Our actions for ambient's movement, unlike those in SAP, report the name of the migrating ambient. For instance, in k.**`enter_`**n we say that ambient k enters n. The knowledge of k is necessary to make the action observable for the environment. This**

[7] L. Cardelli and A. Gordon. Mobile ambients. *heoret ca Co puter c ence*, 240(1):177–213, 2000. An extended abstract appeared in *Proc of Fo aC* .

[8] G. Castagna and F. Zappa Nardelli. The seal calculus revisited: Contextual equivalence and bisimilarity. In *Proc nd F C* , volume 2556 of *LNC* . Springer-Verlag, 2002.

[9] R. De Nicola and M. Hennessy. Testing equivalences for processes. *heoret ca Co puter c ence*, 34:83–133, 1984.

[10] G. Ferrari, U. Montanari, and E. Tuosto. A LTS semantics of ambients via graph synchronization with mobility. In *Proc IC C* , volume 2202 of *LNC* , 2001.

[11] C. Fournet and G. Gonthier. A hierarchy of equivalences for asynchronous calculi. In *Proc th ICALP*, pages 844–855, 1998.

[12] J.C. Godskesen, T. Hildebrandt, and V. Sassone. A calculus of mobile resources. In *Proc th CONC* , volume 2421 of *LNC* , 2002.

[13] A. D. Gordon and L. Cardelli. Equational properties of mobile ambients. *Journa of Mathe at ca tructures n Co puter c ence*, 12:1–38, 2002.

[14] M. Hennessy and J. Riely. A typed language for distributed mobile processes. In *Proc th POPL*. ACM Press, 1998.

[23] D.M. Park. Concurrency on automata and infinite sequences. In P. Deussen, editor, *Conf on heoret ca Co puter c ence*, volume 104 of *LNC* . Springer Verlag, 1981.

[24] D. Sangiorgi. *Express ng Mob ty n Process A gebras F rst Order and H gher Order Parad g s*. PhD thesis CST–99–93, Department of Computer Science, University of Edinburgh, 1992.

[25] D. Sangiorgi. Bisimulation for Higher-Order Process Calculi. *Infor at on and Co pu tat on*, 131(2):141–178, 1996.

[26] D. Sangiorgi. Locality and non-interleaving semantics in calculi for mobile processes. *heoret ca Co puter c ence*, 155:39–83, 1996.

[27] D. Sangiorgi. On the bisimulation proof method. *Journa of Mathe at ca tructures n Co puter c ence*, 8:447–479, 1998.

[28] D. Sangiorgi. Extensionality and intensionality of the ambient logic. In *Proc th POPL*. ACM Press, 2001.

[29] D. Sangiorgi and R. Milner. The problem of “Weak Bisimulation up to”. In *Proc CONC* , volume 630 of *LNC* , pages 32–46. Springer Verlag, 1992.