Università di Pisa

Dipartimento di Informatica

Technical Report : TR-97-16

A Labelled Transition System for pi_epsilon-Calculus

Franck van Breugel

September 09, 1997

ADDR: Corso Italia 40,56125 Pisa,Italy. TEL: +39-50-887111. FAX: +39-50-887226

A Labelled Transition System for π_ϵ-Calculus

Franck van Breugel

Università di Pisa, Department of Computer Science
Corso Italia 40, 56125 Pisa, Italy

franck@di.unipi.it

Abstract

A labelled transition system is presented for Milner's π_ϵ-calculus. This system is related to the reduction system for the calculus presented by Bellin and Scott. Also a reduction system and a labelled transition system for π_ϵI-calculus are given and their correspondence is studied. This calculus is a subcalculus of π_ϵ-calculus in the way Sangiorgi's πI-calculus is a subcalculus of ordinary π-calculus.

Introduction

In the early nineties, Abramsky [Abr94] presented a translation from proofs in linear logic into π-calculus, and outlined the results relating the computational behaviour of the proofs under cut-elimination to that of the processes under reductions. When Milner heard of Abramsky's result, he worked out his own translation. This led to the development of a synchronous version of π-calculus [Mil93], which we call *π_ϵ-calculus*[1]. In [BS94], Bellin and Scott analyzed Abramsky's translation in detail for Milner's π_ϵ-calculus.

In π_ϵ-calculus we encounter enabling, extended scope extrusion, and self communication. These three features are not present in ordinary π-calculus. We discuss them in the following three paragraphs.

In π-calculus, the process $\alpha.P$ specifies that the action α has to precede all actions in P. For the π_ϵ-calculus process $\alpha\, P$ this condition has been weakened as follows. The action α only has to precede those actions in P which it *enables*, i.e. those actions a free name of which is bound by α. For example, in the process $w(x)\,\bar{y}z\,P$, where $x \neq y$, z, the action $w(x)$ does not enable $\bar{y}z$. As a consequence, the action $\bar{y}z$ may precede $w(x)$. Hence, if we put the process $w(x)\,\bar{y}z\,P$ in parallel with $y(z)\,Q$, then a communication at y can occur resulting in the process $w(x)\,P$ in parallel with Q. This is modelled by the reduction

$$w(x)\,\bar{y}z\,P \mid y(z)\,Q \rightarrow w(x)\,P \mid Q. \tag{1}$$

Like in π-calculus, in π_ϵ-calculus we encounter scope extrusions. For example, if $x \neq y$ then

$$(\nu x)\,\bar{y}x\,P \mid y(x)\,Q \rightarrow (\nu x)\,(P \mid Q). \tag{2}$$

Usually, only scopes of the form (νx) are extruded. In π_ϵ-calculus also *extended scopes* like $(\nu w)\,w(x)\,x(y)$—a formal definition of these extended scopes is given in Definition 6—are extruded. For example,

$$(\nu w)\,w(x)\,x(y)\,\bar{z}y\,P \mid z(y)\,Q \rightarrow (\nu w)\,w(x)\,x(y)\,(P \mid Q) \tag{3}$$

provided that $z \neq w$, x, y, and w, x do not occur free in Q.

In π_ϵ-calculus, a process can communicate with itself. In its simplest form, *self communication* amounts to

$$\bar{x}y\,x(z)\,P \rightarrow P[y/z]. \tag{4}$$

[1] Since we do not want to contrast the calculus with *asynchronous* π-calculus [HT91, Bou92] and enablement is one of its key features, we call it π_ϵ-calculus.

Self communication can also take place in extended scopes. For example, if $w \neq x$, y, z then

$$w(x)\,(\nu y)\,y(z)\,\bar{w}zP \rightarrow (\nu y)\,y(z)\,(P[{}^{z}\!/\!_{x}]). \tag{5}$$

The process communicates with itself at w within the extended scope $(\nu y)\,y(z)$.

For π_ϵ-calculus, Bellin and Scott [BS94] presented a *reduction system* [Mil92]. We briefly review this system in Section 2. The rules defining this system are simple and natural. However, the system does not support reasoning in a purely structural way. In Section 3, we give a *labelled transition system* for the calculus following Plotkin's structural approach [Plo81]. The rules defining the labelled transition system are non-trivial. In Section 4, the correctness of this system is shown by proving the correspondence between the reduction system and the labelled transition system. Both the reduction system and the labelled transition system are useful (cf. [San92, page 26]) and once their relation has been established they support each other.

In [San96a], Sangiorgi studied a subcalculus of π-calculus, called πI-calculus, which only uses *internal mobility*. In Section 5, we present a reduction system and a labelled transition system for π_ϵI-calculus, a subcalculus of π_ϵ-calculus with only internal mobility. Furthermore, we investigate the relation between the two systems.

Some related work is discussed in Section 6. In the final section, some conclusions are drawn. We assume that the reader is familiar with π-calculus and πI-calculus. For an introduction to π-calculus we refer the reader to Milner's tutorial [Mil91]. In Sangiorgi's [San96a], πI-calculus is studied in great detail.

Acknowledgements

I am thankful to Prakash Panangaden for numerous discussions. These were essential for my understanding of π_ϵ-calculus and the development of the labelled transition system. Furthermore, I am grateful to Davide Sangiorgi for his constructive comments. My thanks also to Gianluigi Ferrari, Vincent van Oostrom, Marco Pistore, and Philip Scott for discussion.

1 Basic π_ϵ-calculus

We assume an infinite set of *names*. We use x, y, x_1, y', ... to range over these names.

Definition 1 The set of *processes* is defined by

$$P ::= 0 \mid \pi\,P \mid P \mid Q$$

where the set of *particles* is given by

$$\pi ::= \bar{x}y \mid x(y) \mid (\nu x)$$

┘

Only the constructs $\bar{x}y\,P$ and $x(y)\,P$ are not part of ordinary π-calculus. Just a small fragment of π_ϵ-calculus is presented here. We are confident that the results of the present paper can be extended straightforwardly if we add operators like summation and replication.

This calculus has two binders, the particles $x(y)$ and (νx). We define the *bound names* and *free names* of particles and processes in the usual way.

π	bn (π)	fn (π)
$\bar{x}y$	$\emptyset$	$\{x, y\}$
$x(y)$	$\{y\}$	$\{x\}$
(νx)	$\{x\}$	$\emptyset$

P	$\mathrm{bn}\,(P)$	$\mathrm{fn}\,(P)$
0	$\emptyset$	$\emptyset$
$\pi\,P$	$\mathrm{bn}\,(\pi) \cup \mathrm{bn}\,(P)$	$\mathrm{fn}\,(\pi) \cup (\mathrm{fn}\,(P) \setminus \mathrm{bn}\,(\pi))$
$P \mid Q$	$\mathrm{bn}\,(P) \cup \mathrm{bn}\,(Q)$	$\mathrm{fn}\,(P) \cup \mathrm{fn}\,(Q)$

The *names* of particles and processes are given by

$$\begin{aligned} \mathrm{n}\,(\pi) &= \mathrm{bn}\,(\pi) \cup \mathrm{fn}\,(\pi) \\ \mathrm{n}\,(P) &= \mathrm{bn}\,(P) \cup \mathrm{fn}\,(P) \end{aligned}$$

2 Reduction system

The reduction system is defined in two steps. First, we identify several processes by introducing a structural congruence over processes. Second, we define the computation steps of processes in terms of a reduction relation. Our presentation is based on [Mil91, Section 2] and [BS94, Section 2].

Definition 2 The *structural congruence* $\equiv$ is defined as the smallest congruence relation over processes satisfying

1. if P and Q are alpha-convertible then $P \equiv Q$
2. $P \mid Q \equiv Q \mid P$
3. $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$
4. $0 \mid P \equiv P$
5. if $\mathrm{n}\,(\pi_1) \cap \mathrm{bn}\,(\pi_2) = \emptyset$ and $\mathrm{n}\,(\pi_2) \cap \mathrm{bn}\,(\pi_1) = \emptyset$ then $\pi_1\,\pi_2\,P \equiv \pi_2\,\pi_1\,P$
6. if $\mathrm{bn}\,(\pi) \cap \mathrm{fn}\,(Q) = \emptyset$ then $\pi\,(P \mid Q) \equiv (\pi\,P) \mid Q$

$\lrcorner$

For ordinary π-calculus 1., 2., 3., and 4., and 5. and 6. restricted to particles of the form (νx) are used (see [Mil91, page 7 and 8]). In [Eng96, page 81], Engelfriet considers the following variation of 5.

if $x \notin \mathrm{n}\,(\pi)$ then $(\nu x).\pi.P \equiv \pi.(\nu x).P$

Definition 3 The *reduction relation* $\rightarrow$ is defined as the smallest relation over processes satisfying

1. $x(y)\,P \mid \bar{x}z\,Q \rightarrow P[z/y] \mid Q$
2. $\dfrac{P \rightarrow P'}{\pi\,P \rightarrow \pi\,P'}$
3. $\dfrac{P \rightarrow P'}{P \mid Q \rightarrow P' \mid Q}$
4. $\dfrac{P \equiv Q \qquad Q \rightarrow Q' \qquad Q' \equiv P'}{P \rightarrow P'}$

$\lrcorner$

For ordinary π-calculus one only needs 1., 2. restricted to particles of the form (νx), 3., and 4. (see [Mil91, page 8]).

In Appendix A we give proofs of the reductions presented in the introduction. We conclude this section with some properties of the structural congruence. These will be exploited when we link the reduction system and the labelled transition system.

PROPOSITION 4 *If* $P \equiv Q$ *then* $\mathrm{fn}(P) = \mathrm{fn}(Q)$.

PROOF Induction on the proof of $P \equiv Q$. □

PROPOSITION 5 *If* $P \equiv Q$ *then* $P[x/y] \equiv Q[x/y]$.

PROOF Induction on the proof of $P \equiv Q$. □

In Proposition 7 we show that 5. and 6. of Definition 1 also hold for scopes.

DEFINITION 6 The set of *connected input sequences* is given by

$$\iota_x^y ::= x(y) \mid x(z)\,\iota_z^y$$

The set of *scopes* is defined by

$$\sigma_x ::= \iota_y^x \mid (\nu x) \mid (\nu y)\,\iota_y^x$$

⌟

A connected input sequence $\iota_{x_1}^{x_n}$ is of the form

$$x_1(x_2)\,x_2(x_3)\ldots x_{n-1}(x_n).$$

These are related to Sangiorgi's dependency chains [San96a, Definition 6.5]. In ordinary π-calculus one usually only considers scopes of the form (νx). The role of these extended scopes will be discussed in the next section. The bound and free names of scopes are defined straightforwardly.

ι	$\mathrm{bn}(\iota)$	$\mathrm{fn}(\iota)$
$x(y)$	$\{y\}$	$\{x\}$
$x(z)\,\iota_z^y$	$\{z\} \cup \mathrm{bn}(\iota_z^y)$	$\{x\}$

σ	$\mathrm{bn}(\sigma)$	$\mathrm{fn}(\sigma)$
ι_y^x	$\mathrm{bn}(\iota_y^x)$	$\mathrm{fn}(\iota_y^x)$
(νx)	$\{x\}$	$\emptyset$
$(\nu y)\,\iota_y^x$	$\{y\} \cup \mathrm{bn}(\iota_y^x)$	$\emptyset$

PROPOSITION 7

1. *If* $\mathrm{n}(\sigma) \cap \mathrm{bn}(\pi) = \emptyset$ *and* $\mathrm{n}(\pi) \cap \mathrm{bn}(\sigma) = \emptyset$ *then* $\sigma\,\pi\,P \equiv \pi\,\sigma\,P$.
2. *If* $\mathrm{bn}(\sigma) \cap \mathrm{fn}(Q) = \emptyset$ *then* $\sigma\,(P \mid Q) \equiv (\sigma\,P) \mid Q$.

PROOF Structural induction on σ. □

3 Labelled transition system

The labelled transition system presented in this section is new. Its presentation is based on [MPW92, page 46] and [ACS96, page 150]. The system uses the late scheme of name instantiation. It can be adapted straightforwardly to deal with the early scheme (cf. [MPW93]).

The labelled transition system not only describes the computation steps of processes but also their communications possibilities. This information is recorded by means of actions.

DEFINITION 8 The set of *actions* is given by

$$\alpha ::= \bar{x}y \mid x(y) \mid \tau \mid \sigma_y\,\bar{x}y$$

where $x \notin \mathrm{bn}(\sigma_y)$. ⌟

In ordinary π-calculus the action $(\nu y)\,\bar{x}y$ is usually written as $\bar{x}(y)$. The actions $\sigma_y\,\bar{x}y$, with $\sigma_y \neq (\nu y)$, one does not encounter in the usual labelled transition system. These extended scopes are used to model extended scope extrusions (cf. (3) in the introduction).

The bound and free names of actions are defined as follows.

α	$\mathrm{bn}\,(\alpha)$	$\mathrm{fn}\,(\alpha)$
$\bar{x}y$	$\emptyset$	$\{x, y\}$
$x(y)$	$\{y\}$	$\{x\}$
τ	$\emptyset$	$\emptyset$
$\sigma_y\,\bar{x}y$	$\mathrm{bn}\,(\sigma_y)$	$\{x\}$

In the next definition the transition relation is presented. We have omitted the symmetric versions of the rules 9., 10., and 11.

Definition 9 The *transition relation* $\longrightarrow$ is defined as the smallest labelled relation over processes satisfying

1. $\dfrac{P' \xrightarrow{\alpha} Q'}{P \xrightarrow{\alpha} Q}$ $\quad P$ and P', and Q and Q' are alpha-convertible

2. $\bar{x}y\,P \xrightarrow{\bar{x}y} P$

3. $x(y)\,P \xrightarrow{x(y)} P$

4. $\dfrac{P \xrightarrow{\alpha} P'}{\pi\,P \xrightarrow{\alpha} \pi\,P'}$ $\quad \mathrm{n}\,(\alpha) \cap \mathrm{bn}\,(\pi) = \emptyset$ and $\mathrm{n}\,(\pi) \cap \mathrm{bn}\,(\alpha) = \emptyset$

5. $\dfrac{P \xrightarrow{\bar{z}y} P'}{x(y)\,P \xrightarrow{x(y)\,\bar{z}y} P'}$ $\quad y \neq z$ $\qquad \dfrac{P \xrightarrow{\iota_y^w\,\bar{z}w} P'}{x(y)\,P \xrightarrow{x(y)\,\iota_y^w\,\bar{z}w} P'}$ $\quad y \neq z$

6. $\dfrac{P \xrightarrow{\bar{z}y} P'}{(\nu y)\,P \xrightarrow{(\nu y)\,\bar{z}y} P'}$ $\quad y \neq z$ $\qquad \dfrac{P \xrightarrow{\iota_y^w\,\bar{z}w} P'}{(\nu y)\,P \xrightarrow{(\nu y)\,\iota_y^w\,\bar{z}w} P'}$ $\quad y \neq z$

7. $\dfrac{P \xrightarrow{x(z)} P'}{\bar{x}y\,P \xrightarrow{\tau} P'[{}^{y}\!/_{z}]}$

8. $\dfrac{P \xrightarrow{\bar{x}z} P'}{x(y)\,P \xrightarrow{\tau} P'[{}^{z}\!/_{y}]}$ $\quad y \neq x,\, z$ $\qquad \dfrac{P \xrightarrow{\sigma_z\,\bar{x}z} P'}{x(y)\,P \xrightarrow{\tau} \sigma_z\,(P'[{}^{z}\!/_{y}])}$ $\quad y \notin \mathrm{n}\,(\sigma_z\,\bar{x}z)$

9. $\dfrac{P \xrightarrow{\alpha} P'}{P \mid Q \xrightarrow{\alpha} P' \mid Q}$ $\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(Q) = \emptyset$

10. $\dfrac{P \xrightarrow{x(z)} P' \qquad Q \xrightarrow{\bar{x}y} Q'}{P \mid Q \xrightarrow{\tau} P'[{}^{y}\!/_{z}] \mid Q'}$

11. $\dfrac{P \xrightarrow{x(y)} P' \qquad Q \xrightarrow{\sigma_y\,\bar{x}y} Q'}{P \mid Q \xrightarrow{\tau} \sigma_y\,(P' \mid Q')}$ $\quad \mathrm{bn}\,(\sigma_y) \cap \mathrm{fn}\,(P) = \emptyset$

$\lrcorner$

Some remarks:

* The rules 1., 4. with π of the form (νx), the first part of 6., 9., 10., and 11. with σ_y of the form (νy) are as usual.

* The axioms 2. and 3. are as expected.

* The rule 4. models enabling and corresponds to Definition 2.5.

* The rules 5. and 6. describe *scope opening* (cf. [MPW92, page 48]). Like in ordinary π-calculus, the side condition $y \neq z$ prevents z from becoming bound (cf. Definition 8). The rule 11. handles *scope closing*. Note that the scope σ_y reappears in the conclusion. The side condition $\mathrm{bn}(\sigma_y) \cap \mathrm{fn}(P) = \emptyset$ prevents us from deriving the incorrect transition

  $$(\nu z)\, z(y)\, \bar{x}y\, 0 \mid x(y)\, \bar{z}y\, 0 \xrightarrow{\tau} (\nu z)\, z(y)\, (0 \mid \bar{z}y\, 0).$$

  This transition is incorrect since the free name z in $x(y)\, \bar{z}y\, 0$ is only accidentally the same as the bound name z in $(\nu z)\, z(y)\, \bar{x}y\, 0$. In the third example of Appendix B the interplay between scope opening and scope closing is illustrated.

* The rules 7. and 8. describe self communication. Because of the side condition $y \neq x$, we cannot prove the obviously incorrect transition

  $$x(y)\, \bar{y}z\, 0 \xrightarrow{\tau} 0.$$

  This side condition ensures that the free name x and the bound name y, which can be alpha-converted to x, are not identified. The side condition $y \neq z$ rules out the transition

  $$x(y)\, \bar{x}y\, \bar{x}y\, 0 \xrightarrow{\tau} \bar{x}w\, 0.$$

  This transition is incorrect because the bound name y, which can be alpha-converted to w, becomes free (cf. Proposition 10). The side condition $y \notin \mathrm{n}(\sigma_z\, \bar{x}z)$ prevents us from proving the transition

  $$x(y)\, (\nu y)\, y(z)\, \bar{x}z\, \bar{y}w\, 0 \xrightarrow{\tau} (\nu y)\, y(z)\, \bar{z}w\, 0.$$

  This transition is incorrect since the y in $\bar{y}w$ is bound by (νy) in $x(y)\, (\nu y)\, y(z)\, \bar{x}z\, \bar{y}w\, 0$ whereas the corresponding z in $\bar{z}w$ is bound by $y(z)$ in $(\nu y)\, y(z)\, \bar{z}w\, 0$.

* The following variation of 1. (cf. [San92, page 30]) suffices to prove the results of Section 4.

  $$\frac{P' \xrightarrow{\alpha} Q}{P \xrightarrow{\alpha} Q} \quad P \text{ and } P' \text{ are alpha-convertible}$$

  We have chosen for 1. since it is convenient for proving Proposition 11.

In Appendix B we give proofs of the τ-transitions corresponding to the reductions presented in the introduction. Like in the previous section, we conclude with some properties which are used when we relate the two systems.

PROPOSITION 10 *If* $P \xrightarrow{\alpha} P'$ *then* $\mathrm{fn}(\alpha) \subseteq \mathrm{fn}(P)$ *and* $\mathrm{fn}(P') \subseteq \mathrm{fn}(P) \cup \mathrm{bn}(\alpha)$.

PROOF Induction on the proof of $P \xrightarrow{\alpha} P'$. $\square$

The above result also holds for ordinary π-calculus (see [MPW92, Lemma 1]).

PROPOSITION 11 *If there exists a proof of* $P \xrightarrow{\alpha} Q$ *not containing* $\mathrm{bn}(\alpha')$ *and* $\alpha\, Q$ *and* $\alpha'\, Q'$ *are alpha-convertible*[2] *then* $P \xrightarrow{\alpha'} Q'$.

PROOF Induction on the depth of the proof of $P \xrightarrow{\alpha} Q$ exploiting the following fact. Assume there exists a proof of $P \xrightarrow{\alpha} Q$ not containing x. If $y \notin \mathrm{bn}(\alpha)$ then there is a proof of $P[x/y] \xrightarrow{\alpha[x/y]} Q[x/y]$ of the same depth. Otherwise, there is a proof of $P[x/y] \xrightarrow{\alpha[x/y]} Q$ of the same depth. $\square$

This result is similar to [MPW92, Lemma 2].

4 Correspondence between the systems

The reduction system of Section 2 and the labelled transition system of Section 3 are related in this section. More precisely, reductions and τ-transitions are linked. In Theorem 12 it is shown that every τ-transition is matched by a reduction. Conversely, for every reduction there exists a corresponding τ-transition, as is proved in Theorem 15. The proofs of these results are given in Appendix C.

THEOREM 12

1. *If* $P \xrightarrow{\bar{x}y} P'$ *then* $P \equiv \bar{x}y\, P'$.
2. *If* $P \xrightarrow{x(y)} P'$ *then* $P \equiv x(y)\, P'$.
3. *If* $P \xrightarrow{\sigma_y\, \bar{x}y} P'$ *then* $P \equiv \sigma_y\, \bar{x}y\, P'$.
4. *If* $P \xrightarrow{\tau} P'$ *then* $P \rightarrow P'$.

PROOF See Appendix C. $\square$

The proof of Theorem 15 relies on the following lemma. This lemma is the main technical result of the paper.

LEMMA 13 *Let* $P \equiv Q$.

1. *If* $P \xrightarrow{\alpha} P'$ *then* $Q \xrightarrow{\alpha} Q'$ *for some* Q' *such that* $P' \equiv Q'$.
2. *If* $Q \xrightarrow{\alpha} Q'$ *then* $P \xrightarrow{\alpha} P'$ *for some* P' *such that* $P' \equiv Q'$.

PROOF See Appendix C. $\square$

The definitions of late, early, ground, and open bisimulation for π-calculus [MPW92, San96b] can be adapted straightforwardly to our setting (see Appendix D).

COROLLARY 14 $\equiv$ *is a late, early, ground, and open bisimulation.*

PROOF See Appendix D. $\square$

[2] $\tau\, Q$ and $\tau\, Q'$ are alpha-convertible if Q and Q' are.

We conclude this section with

Theorem 15

1. *If* $P \equiv \bar{x}y\,P'$ *then* $P \xrightarrow{\bar{x}y} P''$ *for some* P'' *such that* $P'' \equiv P'$.

2. *If* $P \equiv x(y)\,P'$ *then* $P \xrightarrow{x(y)} P''$ *for some* P'' *such that* $P'' \equiv P'$.

3. *If* $P \equiv \sigma_y\,\bar{x}y\,P'$ *then* $P \xrightarrow{\sigma_y\,\bar{x}y} P''$ *for some* P'' *such that* $P'' \equiv P'$.

4. *If* $P \longrightarrow P'$ *then* $P \xrightarrow{\tau} P''$ *for some* P'' *such that* $P'' \equiv P'$.

Proof See Appendix C. □

5 Basic π_ϵI-calculus

In this section we restrict our attention to a subcalculus of π_ϵ-calculus which only gives rise to *internal mobility* (see [San96a]) called π_ϵI-calculus. The reduction system of Section 2 is easily adapted. Like for ordinary π-calculus, the labelled transition system for the subcalculus is much simpler than the one for the full calculus given in Section 3. The relation between the two systems is similar to the one presented in Section 4.

The subcalculus is obtained by restricting the set of particles. We do not consider *free* outputs $\bar{x}y$ but only *bound* ones $(\nu y)\,\bar{x}y$, from now on abbreviated to $\bar{x}(y)$.

Definition 16 The set of *particles* is given by

$$\pi ::= \bar{x}(y) \mid x(y) \mid (\nu x)$$

⌟

The particle $\bar{x}(y)$ is a binder with

$$\begin{aligned} \mathrm{bn}\,(\bar{x}(y)) &= \{y\} \\ \mathrm{fn}\,(\bar{x}(y)) &= \{x\} \end{aligned}$$

The structural congruence $\equiv$ is defined by all rules of Definition 2 but the rule 5. The latter rule can be derived from the other ones.

Proposition 17 *If* $\mathrm{n}\,(\pi_1) \cap \mathrm{bn}\,(\pi_2) = \emptyset$ *and* $\mathrm{n}\,(\pi_2) \cap \mathrm{bn}\,(\pi_1) = \emptyset$ *then* $\pi_1\,\pi_2\,P \equiv \pi_2\,\pi_1\,P$.

Proof See Appendix C. □

The reduction relation is presented in

Definition 18 The *reduction relation* $\longrightarrow$ is defined as the smallest relation over processes satisfying

1. $x(y)\,P \mid \bar{x}(y)\,Q \longrightarrow (\nu y)\,(P \mid Q)$

2. $\dfrac{P \longrightarrow P'}{\pi\,P \longrightarrow \pi\,P'}$

3. $\dfrac{P \longrightarrow P'}{P \mid Q \longrightarrow P' \mid Q}$

4. $\dfrac{P \equiv Q \qquad Q \longrightarrow Q' \qquad Q' \equiv P'}{P \longrightarrow P'}$

$\lrcorner$

The only difference with Definition 3 is the axiom 1. Note that we only encounter alpha-conversion and no substitution in the reduction system for π_ϵI-calculus.

In the labelled transition system we do not need the extended scopes of Definition 6 we used in Section 3.

Definition 19 The set of *actions* is given by

$$\alpha ::= \bar{x}(y) \mid x(y) \mid \tau$$

$\lrcorner$

The transition relation is presented next. We have omitted the symmetric versions of the rules 7. and 8.

Definition 20 The *transition relation* $\rightarrow$ is defined as the smallest labelled relation over processes satisfying

1. $\dfrac{P' \xrightarrow{\alpha} Q'}{P \xrightarrow{\alpha} Q}$ $\quad P$ and P', and Q and Q' are alpha-convertible

2. $\bar{x}(y)\, P \xrightarrow{\bar{x}(y)} P$

3. $x(y)\, P \xrightarrow{x(y)} P$

4. $\dfrac{P \xrightarrow{\alpha} P'}{\pi\, P \xrightarrow{\alpha} \pi\, P'}$ $\quad \mathrm{n}\,(\alpha) \cap \mathrm{bn}\,(\pi) = \emptyset$ and $\mathrm{n}\,(\pi) \cap \mathrm{bn}\,(\alpha) = \emptyset$

5. $\dfrac{P \xrightarrow{x(z)} P'}{\bar{x}(y)\, P \xrightarrow{\tau} (\nu y)\,(P'[y/z])}$ $\quad x \neq y$

6. $\dfrac{P \xrightarrow{\bar{x}(z)} P'}{x(y)\, P \xrightarrow{\tau} (\nu y)\,(P'[y/z])}$ $\quad x \neq y$

7. $\dfrac{P \xrightarrow{\alpha} P'}{P \mid Q \xrightarrow{\alpha} P' \mid Q}$ $\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(Q) = \emptyset$

8. $\dfrac{P \xrightarrow{\bar{x}(y)} P' \qquad Q \xrightarrow{x(y)} Q'}{P \mid Q \xrightarrow{\tau} (\nu y)\,(P' \mid Q')}$

$\lrcorner$

Some remarks:

* The rules 1., 4., and 7., and the axiom 3. correspond to the rules 1., 4., and 9., and the axiom 3. of Definition 9.

* The axiom 2. and the rules 5., 6., and 8. are the obvious modifications of the axiom 2., and the rules 7., 8., and 10. of Definition 9.

* Note that we do use substitution in the rules 5. and 6. In the transition corresponding to the reduction

$$\bar{x}(y)\, x(z)\, z(w)\, \bar{y}(w)\, 0 \rightarrow (\nu y)\, y(w)\, \bar{y}(w)\, 0 \tag{6}$$

(a proof of this reduction is given in Appendix A) z in $z(w)$ and y in $\bar{y}(w)$ are identified:

$$\bar{x}(y)\, x(z)\, z(w)\, \bar{y}(w)\, 0 \xrightarrow{\tau} (\nu y)\, y(w)\, \bar{y}(w)\, 0.$$

This identification cannot be brought about by alpha-conversion of $\bar{x}(y)\, x(z)\, z(w)\, \bar{y}(w)\, 0$.

We conclude this section with two correspondence theorems.

THEOREM 21

1. *If* $P \xrightarrow{\bar{x}(y)} P'$ *then* $P \equiv \bar{x}(y)\, P'$.

2. *If* $P \xrightarrow{x(y)} P'$ *then* $P \equiv x(y)\, P'$.

3. *If* $P \xrightarrow{\tau} P'$ *then* $P \rightarrow P'$.

PROOF Similar to the proof of Theorem 12. □

THEOREM 22

1. *If* $P \equiv \bar{x}(y)\, P'$ *then* $P \xrightarrow{\bar{x}(y)} P''$ *for some* P'' *such that* $P'' \equiv P'$.

2. *If* $P \equiv x(y)\, P'$ *then* $P \xrightarrow{x(y)} P''$ *for some* P'' *such that* $P'' \equiv P'$.

3. *If* $P \rightarrow P'$ *then* $P \xrightarrow{\tau} P''$ *for some* P'' *such that* $P'' \equiv P'$.

PROOF Similar to the proof of Theorem 15. □

6 Related work

The only three other papers which discuss the relation between a reduction system and a labelled transition system we are aware of are Milner's [Mil92], Honda and Yoshida's [HY93], and Corradini, Ferrari, and Pistore's [CFP97]. All use the rule

$$1.' \quad \frac{P' \xrightarrow{\alpha} Q'}{P \xrightarrow{\alpha} Q} \qquad P \equiv P' \text{ and } Q \equiv Q'$$

instead of

$$1. \quad \frac{P' \xrightarrow{\alpha} Q'}{P \xrightarrow{\alpha} Q} \qquad P \text{ and } P', \text{ and } Q \text{ and } Q' \text{ are alpha-convertible}$$

as we do. In their setting Lemma 13, the main technical result of this paper, becomes trivial. Their rule is less structural than ours. Furthermore, the rule 1. can easily be distributed over the other axioms and rules (compare the labelled transition system of Milner et al. [MPW92, page 46] and the one of Sangiorgi [San92, page 30]). This is not the case for the other rule.

In the conclusion of [MP95], Montanari and Pistore consider relaxing the sequencing power of prefixing. Instead of a reduction system or a labelled transition system, they use a graph rewriting system. In their setting, enablement can easily be accommodated (as long as one does not consider replication).

Conclusion

From our case study we can conclude that the problem of reconstructing a labelled transition system from a reduction system is far from easy. Although the reduction system for π_ϵ-calculus is rather close to the one for ordinary π-calculus, we encounter in the labelled transition system for π_ϵ-calculus extended scopes and various new rules.

In [Mil93, page 37], Milner first presented π_ϵ-calculus with enablement as its new feature. The fact that π_ϵ-calculus has self communication was already observed by Bellin and Scott [BS94, page 15]. But the presence of extended scope extrusion in π_ϵ-calculus—although maybe not very surprising—only occurred to us when we developed the labelled transition system.

The labelled transition system for π_ϵ-calculus might be the basis for the development of a (possibly fully abstract with respect to some form of bisimulation) denotational semantics for the calculus. Here we can make fruitful use of the work of Fiore, Moggi, and Sangiorgi [FMS96], Hennessy [Hen96], and Stark [Sta96].

From Corollary 14 we can conclude that the structural congruence $\equiv$ is indeed included in several well-known behavioural equivalences (this is one of the criteria such a structural congruence should meet [San92, page 27]).

The labelled transition system for π_ϵI-calculus, the subcalculus with only internal mobility, is much simpler than the one for the full calculus. This provides another indication that external mobility is responsible for much of the semantic complications (cf. [San96a]).

Although we only consider internal mobility in π_ϵI-calculus, we do use substitution in the labelled transition system. In πI-calculus only alpha-conversion is needed (see [San96a, Section 2.2]). This suggests that the absence of substitution in πI-calculus is just a property of the calculus, rather than a consequence of its restriction to internal mobility. Whether the substitutions used in π_ϵI-calculus are of a special kind (the substituted name is always bound by a generated restriction) needs further study.

Another topic reserved for later treatment is the study of bisimulation. The definitions of barbed, early, ground, late, and open bisimulation for π-calculus can be adapted straightforwardly to our setting (see Appendix D). We are interested in the connection with bisimulation for action structures (for π_ϵ-calculus) given by Milner in [Mil93].

References

[Abr94] S. Abramsky. Proofs as Processes. *Theoretical Computer Science*, 135(1):5–9, December 1994.

[ACS96] R.M. Amadio, I. Castellani, and D. Sangiorgi. On Bisimulation for the Asynchronous π-Calculus. In U. Montanari and V. Sassone, editors, *Proceedings of CONCUR'96*, volume 1119 of *Lecture Notes in Computer Science*, pages 147–162, Pisa, August 1996. Springer-Verlag.

[Bou92] G. Boudol. Asynchrony and the π-Calculus (note). Report RR-1702, INRIA, Sophia Antipolis, May 1992.

[BS94] G. Bellin and P. Scott. On the π-Calculus and Linear Logic. *Theoretical Computer Science*, 135(1):11–65, December 1994.

[CFP97] F. Corradini, G. Ferrari, and M. Pistore. Eager, Busy-Waiting and Lazy Actions in Timed Computation. In *Proceedings of the 4th Workshop on Expressiveness in Concurrency*, Electronic Notes in Theoretical Computer Science, Santa Margherita Ligure, September 1997. Elsevier Science.

[Eng96] J. Engelfriet. A Multiset Semantics for the π-Calculus. *Theoretical Computer Science*, 153(1/2):65–94, January 1996.

[FMS96] M. Fiore, E. Moggi, and D. Sangiorgi. A Fully Abstract Model for the π-Calculus. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science*, pages 43–54, New Brunswick, July 1996. IEEE Computer Society Press.

[Hen96] M. Hennessy. A Fully Abstract Denotational Semantics for the π-Calculus. Report 96:04, University of Sussex, Brighton, June 1996.

[HT91] K. Honda and M. Tokoro. An Object Calculus for Asynchronous Communication. In P. America, editor, *Proceedings of the European Conference on Object-Oriented Programming*, volume 512 of *Lecture Notes in Computer Science*, pages 133–147, Geneva, July 1991. Springer-Verlag.

[HY93] K. Honda and N. Yoshida. On Reduction-Based Process Semantics. In R.K. Shyamasundar, editor, *Proceedings of the 13th Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 761 of *Lecture Notes in Computer Science*, pages 371–387, Bombay, December 1993. Springer-Verlag.

[Mil80] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1980.

[Mil91] R. Milner. The Polyadic π-Calculus: a tutorial. Report ECS-LFCS-91-180, University of Edinburgh, Edinburgh, October 1991.

[Mil92] R. Milner. Functions as Processes. *Mathematical Structures in Computer Science*, 2(2):119–141, June 1992.

[Mil93] R. Milner. Action Structures for the π-Calculus. Report ECS-LFCS-93-264, University of Edinburgh, Edinburgh, May 1993.

[MP95] U. Montanari and M. Pistore. Concurrent Semantics for the π-Calculus. In S. Brookes, M. Main, A. Melton, and M. Mislove, editors, *Proceedings of the 11th Annual Conference on Mathematical Foundations of Programming Semantics*, volume 1 of *Electronic Notes in Theoretical Computer Science*, New Orleans, March/April 1995. Elsevier Science.

[MPW92] R. Milner, J. Parrow, and D. Walker. A Calculus of Mobile Processes, I and II. *Information and Computation*, 100(1):1–40 and 41–77, September 1992.

[MPW93] R. Milner, J. Parrow, and D. Walker. Modal Logics for Mobile Processes. *Theoretical Computer Science*, 114(1):149–171, June 1993.

[MS92] R. Milner and D. Sangiorgi. Barbed Bisimulation. In W. Kuich, editor, *Proceedings of the 19th International Colloquium on Automata, Languages and Programming*, volume 623 of *Lecture Notes in Computer Science*, pages 685–695, Vienna, July 1992. Springer-Verlag.

[Par81] D. Park. Concurrency and Automata on Infinite Sequences. In P. Deussen, editor, *Proceedings of 5th GI-Conference on Theoretical Computer Science*, volume 104 of *Lecture Notes in Computer Science*, pages 167–183, Karlsruhe, March 1981. Springer-Verlag.

[Plo81] G.D. Plotkin. A Structural Approach to Operational Semantics. Report DAIMI FN-19, Aarhus University, Aarhus, September 1981.

[San92] D. Sangiorgi. *Expressing Mobility in Process Algebras: first-order and higher-order paradigms.* PhD thesis, University of Edinburgh, Edinburgh, 1992.

[San96a] D. Sangiorgi. π-Calculus, Internal Mobility, and Agent-Passing Calculi. *Theoretical Computer Science*, 167(1/2):235–274, October 1996.

[San96b] D. Sangiorgi. A Theory of Bisimulation for π-Calculus. *Acta Informatica*, 33(1):69–97, February 1996.

[Sta96] I. Stark. A Fully Abstract Domain Model for the π-Calculus. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science*, pages 36–42, New Brunswick, July 1996. IEEE Computer Society Press.

A Some reductions

Proofs of the reductions presented in the introduction and Section 5 are given.

1. Let $x \neq y,\ z$. Since

$$w(x)\,\bar{y}z\,P \mid y(z)\,Q \equiv \bar{y}z\,w(x)\,P \mid y(z)\,Q \quad \text{[Definition 2.5]}$$

and

$$\bar{y}z\,w(x)\,P \mid y(z)\,Q \rightarrow w(x)\,P \mid Q \quad \text{[Definition 3.1]}$$

we can conclude that

$$w(x)\,\bar{y}z\,P \mid y(z)\,Q \rightarrow w(x)\,P \mid Q \quad \text{[Definition 3.4]}$$

2. Let $x \neq y$. Because

$$(\nu x)\,\bar{y}x\,P \mid y(x)\,Q \equiv (\nu x)\,(\bar{y}x\,P \mid y(x)\,Q) \quad \text{[Definition 2.6]}$$

and

$$(\nu x)\,(\bar{y}x\,P \mid y(x)\,Q) \rightarrow (\nu x)\,(P \mid Q) \quad \text{[Definition 3.1 and 3.2]}$$

we have that

$$(\nu x)\,\bar{y}x\,P \mid y(x)\,Q \rightarrow (\nu x)\,(P \mid Q) \quad \text{[Definition 3.4]}$$

3. Let $z \neq w,\ x,\ y$, and assume $w,\ x \notin \mathrm{fn}\,(Q)$. Then

$$(\nu w)\,w(x)\,x(y)\,\bar{z}y\,P \mid z(y)\,Q \equiv (\nu w)\,w(x)\,x(y)\,(\bar{z}y\,P \mid z(y)\,Q) \quad \text{[Definition 2.6]}$$

and

$$(\nu w)\,w(x)\,x(y)\,(\bar{z}y\,P \mid z(y)\,Q) \rightarrow (\nu w)\,w(x)\,x(y)\,(P \mid Q) \quad \text{[Definition 3.1 and 3.2]}$$

Hence,

$$(\nu w)\,w(x)\,x(y)\,\bar{z}y\,P \mid z(y)\,Q \rightarrow (\nu w)\,w(x)\,x(y)\,(P \mid Q) \quad \text{[Definition 3.4]}$$

4. Since

$$\begin{aligned} &\bar{x}y\,x(z)\,P \\ &\quad \equiv \quad \bar{x}y\,(0 \mid x(z)\,P) \quad \text{[Definition 2.4]} \\ &\quad \equiv \quad \bar{x}y\,0 \mid x(z)\,P \quad \text{[Definition 2.6]} \end{aligned}$$

and

$$\bar{x}y\,0 \mid x(z)\,P \rightarrow 0 \mid P[{}^{y}\!/\!{}_{z}] \quad \text{[Definition 3.1]}$$

and

$$0 \mid P[{}^{y}\!/\!{}_{z}] \equiv P[{}^{y}\!/\!{}_{z}] \quad \text{[Definition 2.4]}$$

we can conclude that

$$\bar{x}y\,x(z)\,P \rightarrow P[{}^{y}\!/\!{}_{z}] \quad \text{[Definition 3.4]}$$

5. Let $w \neq x, y, z$. Assume $y', z' \notin \mathrm{n}\,(w(x)\,(\nu y)\,y(z)\,\bar{w}zP)$ and $y' \neq z'$. Then

$$
\begin{aligned}
& w(x)\,(\nu y)\,y(z)\,\bar{w}z\,P \\
& \quad \equiv \quad w(x)\,(\nu y')\,y'(z')\,\bar{w}z'\,P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}] \quad \text{[Definition 2.1]} \\
& \quad \equiv \quad (\nu y')\,w(x)\,y'(z')\,\bar{w}z'\,P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}] \quad \text{[Definition 2.5]} \\
& \quad \equiv \quad (\nu y')\,y'(z')\,w(x)\,\bar{w}z'\,P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}] \quad \text{[Definition 2.5]} \\
& \quad \equiv \quad (\nu y')\,y'(z')\,w(x)\,\bar{w}z'\,(0 \mid P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}]) \quad \text{[Definition 2.4]} \\
& \quad \equiv \quad (\nu y')\,y'(z')\,w(x)\,(\bar{w}z'\,0 \mid P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}]) \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad (\nu y')\,y'(z')\,w(x)\,(P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}] \mid \bar{w}z'\,0) \quad \text{[Definition 2.2]} \\
& \quad \equiv \quad (\nu y')\,y'(z')\,(w(x)\,P[{}^{z'}\!/\!_{z}][{}^{y'}\!/\!_{y}] \mid \bar{w}z'\,0) \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad (\nu y)\,y(z)\,(w(x)\,P \mid \bar{w}z\,0) \quad \text{[Definition 2.1]} \\
& \quad \equiv \quad (\nu y)\,y(z)\,(\bar{w}z\,0 \mid w(x)\,P) \quad \text{[Definition 2.2]}
\end{aligned}
$$

Furthermore,

$$(\nu y)\,y(z)\,(\bar{w}z\,0 \mid w(x)\,P) \longrightarrow (\nu y)\,y(z)\,(0 \mid P[{}^{z}\!/\!_{x}]) \quad \text{[Definition 3.1 and 3.2]}$$

and

$$(\nu y)\,y(z)\,(0 \mid P[{}^{z}\!/\!_{x}]) \equiv (\nu y)\,y(z)\,(P[{}^{z}\!/\!_{x}]) \quad \text{[Definition 2.4]}$$

Hence,

$$w(x)\,(\nu y)\,y(z)\,\bar{w}zP \longrightarrow (\nu y)\,y(z)\,(P[{}^{z}\!/\!_{x}]) \quad \text{[Definition 3.4]}$$

6. Since

$$
\begin{aligned}
& \bar{x}(y)\,x(z)\,z(w)\,\bar{y}(w)\,0 \\
& \quad \equiv \quad \bar{x}(y)\,x(z)\,z(w)\,(0 \mid \bar{y}(w)\,0) \quad \text{[Definition 2.4]} \\
& \quad \equiv \quad \bar{x}(y)\,x(z)\,(z(w)\,0 \mid \bar{y}(w)\,0) \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad \bar{x}(y)\,(x(z)\,z(w)\,0 \mid \bar{y}(w)\,0) \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad \bar{x}(y)\,(\bar{y}(w)\,0 \mid x(z)\,z(w)\,0) \quad \text{[Definition 2.2]} \\
& \quad \equiv \quad \bar{x}(y)\,\bar{y}(w)\,0 \mid x(z)\,z(w)\,0 \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad \bar{x}(y)\,\bar{y}(w)\,0 \mid x(y)\,y(w)\,0 \quad \text{[Definition 2.1]}
\end{aligned}
$$

and

$$\bar{x}(y)\,\bar{y}(w)\,0 \mid x(y)\,y(w)\,0 \longrightarrow (\nu y)\,(\bar{y}(w)\,0 \mid y(w)\,0) \quad \text{[Definition 18.1]}$$

and

$$
\begin{aligned}
& (\nu y)\,(\bar{y}(w)\,0 \mid y(w)\,0) \\
& \quad \equiv \quad (\nu y)\,\bar{y}(w)\,(0 \mid y(w)\,0) \quad \text{[Definition 2.6]} \\
& \quad \equiv \quad (\nu y)\,\bar{y}(w)\,y(w)\,0 \quad \text{[Definition 2.4]}
\end{aligned}
$$

we can conclude that

$$\bar{x}(y)\,x(z)\,z(w)\,\bar{y}(w)\,0 \longrightarrow (\nu y)\,\bar{y}(w)\,y(w)\,0 \quad \text{[Definition 18.4]}$$

B Some labelled transitions

Proofs of the τ-transitions corresponding to the reductions presented in the introduction are given.

1. Let $x \neq y,\ z$. Then

$$\dfrac{\dfrac{\bar{y}z\,P \xrightarrow{\bar{y}z} P}{w(x)\,\bar{y}z\,P \xrightarrow{\bar{y}z} w(x)\,P} \qquad y(z)\,Q \xrightarrow{y(z)} Q}{w(x)\,\bar{y}z\,P \mid y(z)\,Q \xrightarrow{\tau} w(x)\,P \mid Q}$$

2. Let $x \neq y$. Then

$$\dfrac{\dfrac{\bar{y}x\,P \xrightarrow{\bar{y}x} P}{(\nu x)\,\bar{y}x\,P \xrightarrow{(\nu x)\,\bar{y}x} P} \qquad y(x)\,Q \xrightarrow{y(x)} Q}{(\nu x)\,\bar{y}x\,P \mid y(x)\,Q \xrightarrow{\tau} (\nu x)\,(P \mid Q)}$$

3. Let $z \neq w,\ x,\ y$, and assume $w,\ x \notin \mathrm{n}\,(Q)$ Then

$$\dfrac{\dfrac{\dfrac{\dfrac{\bar{z}y\,P \xrightarrow{\bar{z}y} P}{x(y)\,\bar{z}y\,P \xrightarrow{x(y)\,\bar{z}y} P}}{w(x)\,x(y)\,\bar{z}y\,P \xrightarrow{w(x)\,x(y)\,\bar{z}y} P}}{(\nu w)\,w(x)\,x(y)\,\bar{z}y\,P \xrightarrow{(\nu w)\,w(x)\,x(y)\,\bar{z}y} P} \qquad z(y)\,Q \xrightarrow{z(y)} Q}{(\nu w)\,w(x)\,x(y)\,\bar{z}y\,P \mid z(y)\,Q \xrightarrow{\tau} (\nu w)\,w(x)\,x(y)\,(P \mid Q)}$$

4. We have that

$$\dfrac{x(z)\,P \xrightarrow{x(z)} P}{\bar{x}y\,x(z)\,P \xrightarrow{\tau} P[{}^{y}\!/\!{}_{z}]}$$

5. Let $w \neq x,\ y,\ z$. Then

$$\dfrac{\dfrac{\dfrac{\bar{w}z\,P \xrightarrow{\bar{w}z} P}{y(z)\,\bar{w}z\,P \xrightarrow{y(z)\,\bar{w}z} P}}{(\nu y)\,y(z)\,\bar{w}z\,P \xrightarrow{(\nu y)\,y(z)\,\bar{w}z} P}}{w(x)\,(\nu y)\,y(z)\,\bar{w}z\,P \xrightarrow{\tau} (\nu y)\,y(z)\,(P[{}^{z}\!/\!{}_{y}])}$$

C Some proofs

The proofs omitted in Section 4 and 5 are given.

Proof of Theorem 12 This theorem is proved by induction on the proofs. For example, assume the proof is of the form

$$\dfrac{\begin{array}{c}\vdots\\ P \xrightarrow{\sigma_z\,\bar{x}z} P'\end{array}}{x(y)\,P \xrightarrow{\tau} \sigma_z\,(P'[{}^{z}\!/\!{}_{y}])}\quad y \notin \mathrm{n}\,(\sigma_z\,\bar{x}z)$$

By induction, $P \equiv \sigma_z\, \bar{x}z\, P'$. Hence,

$$
\begin{array}{ll}
\multicolumn{2}{l}{x(y)\, P} \\
\equiv & x(y)\, \sigma_z\, \bar{x}z\, P' \\
\equiv & \sigma_z\, x(y)\, \bar{x}z\, P' \quad \text{[Proposition 7.1]} \\
\equiv & \sigma_z\, x(y)\, \bar{x}z\, (0 \mid P') \quad \text{[Definition 2.4]} \\
\equiv & \sigma_z\, x(y)\, (\bar{x}z\, 0 \mid P') \quad \text{[Definition 2.6]} \\
\equiv & \sigma_z\, x(y)\, (P' \mid \bar{x}z\, 0) \quad \text{[Definition 2.2]} \\
\equiv & \sigma_z\, (x(y)\, P' \mid \bar{x}z\, 0) \quad \text{[Definition 2.6]} \\
\equiv & \sigma_z\, (\bar{x}z\, 0 \mid x(y)\, P') \quad \text{[Definition 2.2]}
\end{array}
$$

Furthermore,

$$\sigma_z\, (\bar{x}z\, 0 \mid x(y)\, P') \rightarrow \sigma_z\, (0 \mid P'[z/y]) \quad \text{[Definition 3.1 and 3.2]}$$

and

$$\sigma_z\, (0 \mid P'[z/y]) \equiv \sigma_z\, (P'[z/y]) \quad \text{[Definition 2.4]}$$

Hence, we can conclude that

$$x(y)\, P \rightarrow \sigma_z\, (P'[z/y]) \quad \text{[Definition 3.4]}$$

□

PROOF OF LEMMA 13 We prove this lemma by induction on the proofs of $P \xrightarrow{\alpha} P'$, $Q \xrightarrow{\alpha} Q'$, and $P \equiv Q$. We only consider proofs of $P \xrightarrow{\alpha} P'$ and $Q \xrightarrow{\alpha} Q'$ of minimal complexity. The complexity of a proof is determined by those nodes in the proof where the rule 1. is applied. The more the rule 1. is applied towards the root of the proof, the smaller its complexity is. Only a few of many cases are elaborated on.

- Assume $P \equiv Q$. Consider the proof

$$\frac{\begin{array}{c}\vdots \\ P' \xrightarrow{\alpha} R'\end{array}}{P \xrightarrow{\alpha} R} \quad P \text{ and } P', \text{ and } R \text{ and } R' \text{ are alpha-convertible}$$

  Because P and P' are alpha-convertible, $P \equiv P'$. Since $P \equiv Q$, we have that $P' \equiv Q$. By induction, $Q \xrightarrow{\alpha} Q'$ for some Q' such that $R' \equiv Q'$. Because R and R' are alpha-convertible, we have that $R \equiv R'$. Hence, $R \equiv Q'$.

- Consider the axiom $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$ and the proof

$$\frac{\dfrac{\begin{array}{c}\vdots \\ P' \mid Q' \xrightarrow{\alpha} S'\end{array}}{P \mid Q \xrightarrow{\alpha} S} \quad P' \mid Q' \text{ and } P \mid Q, \text{ and } S \text{ and } S' \text{ are alpha-convertible}}{(P \mid Q) \mid R \xrightarrow{\alpha} S \mid R} \quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(R) = \emptyset$$

Then

$$\cfrac{\cfrac{\begin{array}{c}\vdots\\ P' \mid Q' \xrightarrow{\alpha} S'\end{array}}{(P' \mid Q') \mid R \xrightarrow{\alpha} S' \mid R}\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(R) = \emptyset}{(P \mid Q) \mid R \xrightarrow{\alpha} S \mid R}\quad (P \mid Q) \mid R \text{ and } (P' \mid Q') \mid R\text{, and } S \mid R \text{ and } S' \mid R \text{ are alpha-convertible}$$

and the complexity of this proof is smaller.

* Consider the proofs

$$\cfrac{\begin{array}{c}\vdots\\ P \equiv Q\end{array}}{P \mid R \equiv Q \mid R} \qquad \text{and} \qquad \cfrac{\begin{array}{c}\vdots\\ R \xrightarrow{\alpha} R'\end{array}}{P \mid R \xrightarrow{\alpha} P \mid R'}\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(P) = \emptyset$$

According to Proposition 4, $\mathrm{fn}\,(P) = \mathrm{fn}\,(Q)$. Hence,

$$\cfrac{\begin{array}{c}\vdots\\ R \xrightarrow{\alpha} R'\end{array}}{Q \mid R \xrightarrow{\alpha} Q \mid R'}\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(Q) = \emptyset \qquad \text{and} \qquad \cfrac{\begin{array}{c}\vdots\\ P \equiv Q\end{array}}{P \mid R' \equiv Q \mid R'}$$

* Consider the proofs

$$\cfrac{\begin{array}{c}\vdots\\ P \equiv Q\end{array}}{\bar{x}y\,P \equiv \bar{x}y\,Q} \qquad \text{and} \qquad \cfrac{\begin{array}{c}\vdots\\ P \xrightarrow{x(z)} P'\end{array}}{\bar{x}y\,P \xrightarrow{\tau} P'[y/z]}$$

By induction,

$$\cfrac{\begin{array}{c}\vdots\\ Q \xrightarrow{x(z)} Q'\end{array}}{\bar{x}y\,Q \xrightarrow{\tau} Q'[y/z]}$$

for some Q' such that $P' \equiv Q'$. According to Proposition 5, $P'[y/z] \equiv Q'[y/z]$.

* Assume $\mathrm{bn}\,(\pi) \cap \mathrm{fn}\,(Q) = \emptyset$. Consider the axiom $\pi\,(P \mid Q) \equiv (\pi\,P) \mid Q$ and the proof

$$\cfrac{\cfrac{\begin{array}{c}\vdots\\ Q \xrightarrow{\alpha} Q'\end{array}}{P \mid Q \xrightarrow{\alpha} P \mid Q'}\quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(P) = \emptyset}{\pi\,(P \mid Q) \xrightarrow{\alpha} \pi\,(P \mid Q')}\quad \mathrm{n}\,(\alpha) \cap \mathrm{bn}\,(\pi) = \emptyset \text{ and } \mathrm{n}\,(\pi) \cap \mathrm{bn}\,(\alpha) = \emptyset$$

Then

$$\dfrac{\begin{array}{c}\vdots\\ Q \xrightarrow{\alpha} Q'\end{array}}{(\pi\, P) \mid Q \xrightarrow{\alpha} (\pi\, P) \mid Q'} \quad \mathrm{bn}\,(\alpha) \cap \mathrm{fn}\,(\pi\, P) = \emptyset$$

From Proposition 10 we can deduce that $\mathrm{bn}\,(\pi) \cap \mathrm{fn}\,(Q') = \emptyset$. Hence, $\pi\,(P \mid Q') \equiv (\pi\, P) \mid Q'$.

* Consider the axiom $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$ and the proof

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ P \xrightarrow{\bar{x}y} P'\end{array} \qquad \begin{array}{c}\vdots\\ Q \xrightarrow{x(z)} Q'\end{array}}{P \mid Q \xrightarrow{\tau} P' \mid (Q'[y/z])}}{(P \mid Q) \mid R \xrightarrow{\tau} (P' \mid (Q'[y/z])) \mid R}$$

Let w be such that it does not appear in the proof of $Q \xrightarrow{x(z)} Q'$ and $w \notin \mathrm{fn}\,(R)$. According to Proposition 11,

$$\dfrac{\begin{array}{c}\vdots\\ P \xrightarrow{\bar{x}y} P'\end{array} \qquad \dfrac{\begin{array}{c}\vdots\\ Q \xrightarrow{x(w)} Q'[w/z]\end{array}}{Q \mid R \xrightarrow{x(w)} Q'[w/z] \mid R}\ \mathrm{bn}\,(x(w)) \cap \mathrm{fn}\,(R) = \emptyset}{P \mid (Q \mid R) \xrightarrow{\tau} P' \mid (Q'[w/z] \mid R)[y/w]}$$

and

$$\begin{array}{ll} P' \mid (Q'[w/z] \mid R)[y/w] & \\ \quad \equiv \ P' \mid (Q'[y/z] \mid R) & [w \notin \mathrm{fn}\,(R) \cup \mathrm{n}\,(Q')] \\ \quad \equiv \ (P' \mid Q'[y/z]) \mid R & [\text{Definition 2.3}] \end{array}$$

□

Proof of Theorem 15

1. Obviously, $\bar{x}y\, P' \xrightarrow{\bar{x}y} P'$. Since $P \equiv \bar{x}y\, P'$, we can conclude from Lemma 13 that $P \xrightarrow{\bar{x}y} P''$ for some P'' such that $P'' \equiv P'$.
2. Similar to 1.
3. Similar to 1.
4. We prove this case by induction on the proof of $P \rightarrow P'$. For example, consider the proof

$$\dfrac{\begin{array}{c}\vdots\\ P \equiv Q\end{array} \qquad \begin{array}{c}\vdots\\ Q \rightarrow Q'\end{array} \qquad \begin{array}{c}\vdots\\ Q' \equiv P'\end{array}}{P \rightarrow P'}$$

By induction, $Q \xrightarrow{\tau} Q''$ for some Q'' such that $Q'' \equiv Q'$. According to Lemma 13, $P \xrightarrow{\tau} P''$ for some P'' such that $P'' \equiv Q''$. Consequently, $P'' \equiv P'$.

□

Proof of Proposition 17 One can prove that $P \equiv P_X \mid P_{\bar{X}}$ for some P_X and $P_{\bar{X}}$ such that $\mathrm{fn}\,(P_X) \subseteq X$ and $\mathrm{fn}\,(P_{\bar{X}}) \subseteq \bar{X}$ by structural induction on P.[3] Hence,

$$\begin{array}{ll}
\pi_1\,\pi_2\,P & \\
\quad\equiv \;\; \pi_1\,\pi_2\,(P_{\mathrm{bn}\,(\pi_1)} \mid P_{\overline{\mathrm{bn}\,(\pi_1)}}) & [\text{induction}] \\
\quad\equiv \;\; \pi_1\,(P_{\mathrm{bn}\,(\pi_1)} \mid \pi_2\,P_{\overline{\mathrm{bn}\,(\pi_1)}}) & [\mathrm{fn}\,(P_{\mathrm{bn}\,(\pi_1)}) \subseteq \mathrm{bn}\,(\pi_1) \subseteq \mathrm{n}\,(\pi_1)] \\
\quad\equiv \;\; \pi_1\,P_{\mathrm{bn}\,(\pi_1)} \mid \pi_2\,P_{\overline{\mathrm{bn}\,(\pi_1)}} & [\mathrm{fn}\,(\pi_2\,P_{\overline{\mathrm{bn}\,(\pi_1)}}) \subseteq \mathrm{fn}\,(\pi_2) \cup \overline{\mathrm{bn}\,(\pi_1)} \subseteq \mathrm{n}\,(\pi_2) \cup \overline{\mathrm{bn}\,(\pi_1)}] \\
\quad\equiv \;\; \pi_2\,(\pi_1\,P_{\mathrm{bn}\,(\pi_1)} \mid P_{\overline{\mathrm{bn}\,(\pi_1)}}) & [\mathrm{fn}\,(\pi_1\,P_{\mathrm{bn}\,(\pi_1)}) \subseteq \mathrm{n}\,(\pi_1)] \\
\quad\equiv \;\; \pi_2\,\pi_1\,(P_{\mathrm{bn}\,(\pi_1)} \mid P_{\overline{\mathrm{bn}\,(\pi_1)}}) & [\mathrm{fn}\,(P_{\overline{\mathrm{bn}\,(\pi_1)}}) \subseteq \overline{\mathrm{bn}\,(\pi_1)}] \\
\quad\equiv \;\; \pi_2\,\pi_1\,P & [\text{induction}]
\end{array}$$

□

D Some bisimulations

The definitions of barbed, early, ground, late, and open bisimulation for π-calculus are adapted to our setting. It is also shown that the structural congruence $\equiv$ is an example of all these notions. Further study of these equivalences is left for future research.

D.1 Barbed bisimulation

Barbed bisimulation has been introduced by Milner and Sangiorgi in [MS92]. We adapt this notion to our setting as follows. The *free subject names* of processes are given by

$$\begin{array}{ll}
\mathrm{fsn}\,(0) & = \emptyset \\
\mathrm{fsn}\,(\bar{x}y\,P) & = \{x\} \cup \mathrm{fsn}\,(P) \\
\mathrm{fsn}\,(x(y)\,P) & = \{x\} \cup (\mathrm{fsn}\,(P) \setminus \{y\}) \\
\mathrm{fsn}\,((\nu x)\,P) & = \mathrm{fsn}\,(P) \setminus \{x\} \\
\mathrm{fsn}\,(P \mid Q) & = \mathrm{fsn}\,(P) \cup \mathrm{fsn}\,(Q)
\end{array}$$

Definition 23 A relation $\mathcal{R}$ over processes is a *barbed bisimulation* if $P\,\mathcal{R}\,Q$ implies

* if $P \rightarrow P'$ then there exists a Q' such that $Q \rightarrow Q'$ and $P'\,\mathcal{R}\,Q'$,
* if $Q \rightarrow Q'$ then there exists a P' such that $P \rightarrow P'$ and $P'\,\mathcal{R}\,Q'$, and
* $\mathrm{fsn}\,(P) = \mathrm{fsn}\,(Q)$.

┘

Proposition 24 *$\equiv$ is a barbed bisimulation.*

Proof Let $P \equiv Q$. If $P \rightarrow P'$ then $Q \rightarrow P'$ by Definition 3.4. Furthermore, one can easily check that $P \equiv Q$ implies $\mathrm{fsn}\,(P) = \mathrm{fsn}\,(Q)$ by induction on the proof of $P \equiv Q$. □

[3] By $\bar{X}$ we denote the set-theoretic complement of X.

D.2 Early bisimulation

The definition of early bisimulation given by Milner et al. [MPW92] applies also to our setting.

Definition 25 A relation $\mathcal{R}$ over processes is an *early bisimulation* if $P \mathcal{R} Q$ implies

* if $P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha = x(y)$ then for all z there exists a Q' such that $Q \xrightarrow{\alpha} Q'$ and $P'[z/y] \mathcal{R} Q'[z/y]$,
* if $P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha \neq x(y)$ then there exists a Q' such that $Q \xrightarrow{\alpha} Q'$ and $P' \mathcal{R} Q'$,
* if $Q \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha = x(y)$ then for all z there exists a P' such that $P \xrightarrow{\alpha} P'$ and $P'[z/y] \mathcal{R} Q'[z/y]$, and
* if $Q \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha \neq x(y)$ then there exists a P' such that $P \xrightarrow{\alpha} P'$ and $P' \mathcal{R} Q'$.

┘

Proposition 26 $\equiv$ *is an early bisimulation.*

Proof Immediate consequence of Lemma 13 and Proposition 5. □

D.3 Ground bisimulation

Ordinary bisimulation, a notion due to Milner [Mil80] and Park [Par81], becomes ground bisimulation in the setting of π-calculus.

Definition 27 A relation $\mathcal{R}$ over processes is a *ground bisimulation* if $P \mathcal{R} Q$ implies

* if $P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ then there exists a Q' such that $Q \xrightarrow{\alpha} Q'$ and $P' \mathcal{R} Q'$ and
* if $Q \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ then there exists a P' such that $P \xrightarrow{\alpha} P'$ and $P' \mathcal{R} Q'$.

┘

Proposition 28 $\equiv$ *is a ground bisimulation.*

Proof Immediate consequence of Lemma 13. □

D.4 Late bisimulation

Late bisimulation for ordinary π-calculus as presented by Milner, Parrow, and Walker [MPW92] also applies to our setting.

Definition 29 A relation $\mathcal{R}$ over processes is a *late bisimulation* if $P \mathcal{R} Q$ implies

* if $P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha = x(y)$ then there exists a Q' such that $Q \xrightarrow{\alpha} Q'$ and for all z, $P'[z/y] \mathcal{R} Q'[z/y]$
* if $P \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha \neq x(y)$ then there exists a Q' such that $Q \xrightarrow{\alpha} Q'$ and $P' \mathcal{R} Q'$,
* if $Q \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha = x(y)$ then there exists a P' such that $P \xrightarrow{\alpha} P'$ and for all z, $P'[z/y] \mathcal{R} Q'[z/y]$, and

* if $Q \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q)) = \emptyset$ and $\alpha \neq x(y)$ then there exists a P' such that $P \xrightarrow{\alpha} P'$ and $P' \mathcal{R} Q'$.

$\lrcorner$

PROPOSITION 30 $\equiv$ *is a late bisimulation.*

PROOF Immediate consequence of Lemma 13 and Proposition 5. $\square$

D.5 Open bisimulation

Sangiorgi's open bisimulation [San96b] boils down to the following.

DEFINITION 31 A relation $\mathcal{R}$ over processes is an *open bisimulation* if $P \mathcal{R} Q$ implies that for all substitutions ς,

* if $P\varsigma \xrightarrow{\alpha} P'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q) \cup \mathrm{n}(\varsigma)) = \emptyset$ then there exists a Q' such that $Q\varsigma \xrightarrow{\alpha} Q'$ and $P' \mathcal{R} Q'$ and
* if $Q\varsigma \xrightarrow{\alpha} Q'$ and $\mathrm{bn}(\alpha) \cap (\mathrm{n}(P) \cup \mathrm{n}(Q) \cup \mathrm{n}(\varsigma)) = \emptyset$ then there exists a P' such that $P\varsigma \xrightarrow{\alpha} P'$ and $P' \mathcal{R} Q'$.

$\lrcorner$

PROPOSITION 32 $\equiv$ *is an open bisimulation.*

PROOF Immediate consequence of Lemma 13 and (an obvious generalization of) Proposition 5. $\square$