

A Schematic Axiom for Open Congruence

Yuxi Fu*

Department of Computer Science
Shanghai Jiaotong University, Shanghai 200030, China

Abstract. A schematic law dealing with localization operator is proposed for pi calculus. It is shown that the law renders the use of distinction unnecessary in the axiomatic theory of open congruence.

1 Introduction

The role of bisimulation equivalences in process algebra has been reinforced over the last decades. The studies of the π -calculi ([15]) and many of its variants have demonstrated both the diversity and the pertinent nature of bisimulation equivalences. Bisimulation equivalences are stronger than other observational equivalences in the sense that they are not sensitive to the changes of environment. In other words no matter how environment changes two bisimilar processes can always simulate each other. In CCS ([14]) the bisimulation approach gives rise to a relatively small number of equivalence relations. In the π -calculus however there is a proliferation of bisimulation equivalences, one of which is the open equivalence ([18]). The open equivalence assumes that environments may have significant effect on processes. The change of an environment could modify the names of process residing in the environment although it can not alter the topological structure of the process. This requires that the definition of open bisimulation incorporate the idea that process equivalence should be closed under substitution of names in every bisimulation step. Recent studies on χ -calculus ([1–5, 8–10, 17]) shows that the assumption of open bisimulation is very reasonable.

The open equivalence was proposed by Sangiorgi who did just enough to convince the reader of the simplicity of the relation. Due to the very definition of bisimulation, proofs of theoretical properties of bisimulation equivalences often rely on inductions on bisimulation steps. For mobile processes bisimulations incur name substitutions. Therefore inductions are made easier if the subject bisimulations are closed under substitutions. Take for example Sangiorgi's complete axiomatization system for strong open equivalence. The proof of the completeness theorem is a straightforward induction. It is considerably simpler than the corresponding proofs for strong early equivalence and strong late equivalence using symbolic approach ([11, 13]). However in order to deal with localization Sangiorgi's system introduces distinctions, which is a form of conditions as used in the symbolic framework.

The definition of weak open equivalence, for the π -calculus without the mismatch combinator, can be obtained in the standard manner. Most part of the algebraic theory of this weak relation is simple except in one aspect. Contrary to the popular belief, the complete system for the strong open equivalence extended with Milner's three tau laws does not constitute a complete system for the weak open congruence. A new tau law, T4, proposed by the present author is necessary.

It has to be said that Sangiorgi's definition of weak open bisimulation applies only to the π -calculus without the mismatch operator. It was discovered by Fu and Zhang ([6, 7]) that if one applies Sangiorgi's definition to the π -calculus with the mismatch operator one gets a relation that is not closed under composition! The following example is given in [6]: The processes $a(x).P + a(x).[x \neq y]P$ and $a(x).[x \neq y]P$ should not be bisimilar; but they are indistinguishable by Sangiorgi's original definition. To bypass the problem Fu and Zhang have proposed two new equivalence relations. They are the early open bisimilarity and the late open bisimilarity. The terminologies should suggest that the difference between these two equivalence relations draws a similarity to that between the early equivalence and the late equivalence. Fu and Zhang have investigated axiomatization systems for both early and late open congruences. The two complete systems contain the following schematic laws:

$$(x)C[[x \neq y]P] = (x)C[P] \quad (1)$$

* The author is supported by the NNSFC for Distinguished Young Scholars (60225012) and the Development Fund of Shanghai Committee of Science and Technology (025115032). He is also supported by BASICS, The Laboratory for Basic Studies in Computing Science, sponsored by Shanghai Education Commission. BASICS is affiliated to the Department of Computer Science of Shanghai Jiaotong University.

where x, y are not bound in $C[]$ and $x \neq y$. This law is equivalent to the following schematic law

$$(x)C[[x=y]P] = (x)C[\mathbf{0}] \quad (2)$$

where x, y are not bound in $C[]$ and $x \neq y$. It is clear that (2) is more general than (1) since it does not refer to the mismatch combinator. These schematic laws render distinctions unnecessary.

The purpose of this paper is to show that one can use (2) to give a distinction free complete system for the open congruence of the π -calculus without the mismatch operator. The structure of the paper is as follows: Section 2 introduces the necessary background materials of π -calculus. Section 3 gives the strong and the weak systems. Section 4 defines internal distinction and discusses some properties of it. Section 5 proves a generalized completeness result.

2 Preliminaries

This section fixes up some preliminary notations of π -calculus. The grammar of the π -processes without the mismatch operator is as follows:

$$P := \mathbf{0} \mid \pi.P \mid P \mid P \mid (x)P \mid P+P \mid [x=y]P \mid !P$$

where

$$\pi := a(x) \mid \bar{a}x \mid \tau$$

The set \mathcal{P} of processes is defined in terms of the set \mathcal{N} of names. In both $a(x).P$ and $(x)P$ the name x is bound. The notations $fn(-), bn(-), n(-)$ are standard. A substitution σ is a function from \mathcal{N} to \mathcal{N} such that $\{x \mid \sigma(x) \neq x\}$ is a finite set. We write $P\sigma$ for the process obtained by substituting names in P according to σ . The composition of two substitutions σ, σ' is denoted by $\sigma\sigma'$ and $P\sigma\sigma'$ is understood as $(P\sigma)\sigma'$. We often write $\{y_1/x_1, \dots, y_n/x_n\}$ for the substitution that maps x_1, \dots, x_n respectively onto y_1, \dots, y_n and maps any other name to itself.

The operational semantics is defined in terms of the following labelled transition system:

Prefix

$$\frac{}{\pi.P \xrightarrow{\pi} P}$$

Composition

$$\frac{P \xrightarrow{\lambda} P'}{P \mid Q \xrightarrow{\lambda} P' \mid Q} \quad \frac{P \xrightarrow{a(x)} P' \quad Q \xrightarrow{\bar{a}y} Q'}{P \mid Q \xrightarrow{\tau} P'\{y/x\} \mid Q'} \quad \frac{P \xrightarrow{a(x)} P' \quad Q \xrightarrow{\bar{a}(x)} Q'}{P \mid Q \xrightarrow{\tau} (x)(P' \mid Q')}$$

Restriction

$$\frac{P \xrightarrow{\lambda} P' \quad x \notin n(\lambda)}{(x)P \xrightarrow{\lambda} (x)P'} \quad \frac{P \xrightarrow{\bar{a}x} P'}{(x)P \xrightarrow{\bar{a}(x)} P'}$$

Condition

$$\frac{P \xrightarrow{\lambda} P'}{[x=x]P \xrightarrow{\lambda} P'}$$

Choice

$$\frac{P \xrightarrow{\lambda} P'}{P+Q \xrightarrow{\lambda} P'}$$

Replication

$$\frac{P \mid !P \xrightarrow{\lambda} P'}{!P \xrightarrow{\lambda} P'}$$

The letter λ ranges over the set $\{a(x), \bar{a}x, \bar{a}(x) \mid a, x \text{ are names}\} \cup \{\tau\}$ of action labels. In what follows μ stands for a finite sequence of match operators concatenated one after another. We will write $C[]$ for a context inductively defined as follows: (i) $(-)$ is a context; (ii) if $C[]$ is a context then $C[] \mid P, P \mid C[], (x)C[], C[]+P, P+C[], [x=y]C[], a(x).C[], \bar{a}x.C[]$ and $\tau.C[]$ are contexts.

A nice property about the π -calculus without the mismatch operator is that actions are preserved by substitution of names.

Lemma 1. *If $P \xrightarrow{\lambda} P'$ then $P\sigma \xrightarrow{\lambda\sigma} P'\sigma$ for any substitution σ .*

When defining bisimulation equivalences for π -processes the most subtle problem is to do with the localization operator. Sangiorgi's solution to the problem is to use distinctions.

Definition 2. A distinction is a finite symmetric and irreflexive relation on names. Distinctions will be denoted by D, D' etc.. The notation $D \setminus x$ denotes $D \setminus \{(x, y), (y, x) \mid y \text{ is a name}\}$.

A substitution σ respects a distinction D if $\sigma(x) \neq \sigma(y)$ whenever $(x, y) \in D$. A sequence of match combinators μ respects a distinction D if $\mu \not\equiv x=y$ whenever $(x, y) \in D$.

The strong and weak open bisimilarities are defined as follows.

Definition 3. The set $\mathcal{R} = \{\mathcal{R}^D\}$ of binary symmetric relations on processes is an open bisimulation if for each distinction D and for each substitution σ respecting D , $P\mathcal{R}^D Q$ implies the following properties:

- (i) If $P\sigma \xrightarrow{\lambda} P'$ and λ is not a restricted output action then Q' exists such that $Q\sigma \xrightarrow{\lambda} Q'$ and $P'\mathcal{R}^{D\sigma} Q'$.
- (ii) If $P\sigma \xrightarrow{\bar{a}(x)} P'$ then Q' exists such that $Q\sigma \xrightarrow{\bar{a}(x)} Q'$ and $P'\mathcal{R}^{D'} Q'$, where D' is $D\sigma \cup \{x\} \times \text{fn}(P\sigma + Q\sigma) \cup \text{fn}(P\sigma + Q\sigma) \times \{x\}$.

P is open bisimilar to Q with respect to D , notation $P \sim_o^D Q$, if there exists an open bisimulation \mathcal{R} such that $(P, Q) \in \mathcal{R}^D$. We say that P is open bisimilar to Q if $P \sim_o^\emptyset Q$.

Definition 4. The set $\mathcal{R} = \{\mathcal{R}^D\}$ of binary symmetric relations on processes is a weak open bisimulation if for each distinction D and for each substitution σ respecting D , $P\mathcal{R}^D Q$ implies the following properties:

- (i) If $P\sigma \xrightarrow{\lambda} P'$ and λ is not a restricted output action then Q' exists such that $Q\sigma \xrightarrow{\hat{\lambda}} Q'$ and $P'\mathcal{R}^{D\sigma} Q'$.
- (ii) If $P\sigma \xrightarrow{\bar{a}(x)} P'$ then Q' exists such that $Q\sigma \xrightarrow{\bar{a}(x)} Q'$ and $P'\mathcal{R}^{D'} Q'$, where D' is $D\sigma \cup \{x\} \times \text{fn}(P\sigma + Q\sigma) \cup \text{fn}(P\sigma + Q\sigma) \times \{x\}$.

P is weak open bisimilar to Q with respect to D , notation $P \approx_o^D Q$, if there exists a weak open bisimulation \mathcal{R} such that $(P, Q) \in \mathcal{R}^D$. We say that P is weak open bisimilar to Q if $P \approx_o^\emptyset Q$.

As is well-known \approx_o^D is not a congruence relation. The standard approach to obtain a congruence relation from \approx_o^D is given by the following definition.

Definition 5. P and Q are open congruent, notation $P \simeq_o^D Q$, if $P \approx_o^D Q$ and, for each substitution σ respecting D , the following conditions are satisfied:

- (i) If $P\sigma \xrightarrow{\tau} P'$ then $Q\sigma \xrightarrow{\tau} Q'$ such that $P' \approx_o^D Q'$.
- (ii) If $Q\sigma \xrightarrow{\tau} Q'$ then $P\sigma \xrightarrow{\tau} P'$ such that $P' \approx_o^D Q'$.

Before ending this section we mention some useful results.

Lemma 6. The following properties hold:

- (i) If $P \sim_o^D Q$ then $P \sim_o^{D'} Q$ for each distinction D' such that $D \subseteq D'$.
- (ii) If $P \approx_o^D Q$ then $P \approx_o^{D'} Q$ for each distinction D' such that $D \subseteq D'$.

Lemma 7. The following properties hold:

- (i) If $P \sim_o^D Q$ and $(x, y) \notin D$ then $P\{y/x\} \sim_o^{D\{y/x\}} Q\{y/x\}$.
- (ii) If $P \approx_o^D Q$ and $(x, y) \notin D$ then $P\{y/x\} \approx_o^{D\{y/x\}} Q\{y/x\}$.

Proof. Suppose σ respects $D\{y/x\}$. Then $\{y/x\}\sigma$ respects D . So (i) and (ii) follow from definitions. \square

3 System

A complete system for process congruence consists of a set of equational laws and a set of inference rules. These laws and rules contain the usual ones for equivalence and congruence. Figure 1 contains some useful laws for the strong open congruence. It is well-known that an expansion law is necessary in order to convert processes in parallel composition form to those in summation form. We omit the standard expansion law in this paper as it is well-known. So in the rest of the paper we focus on the fragment of the π -calculus with neither the composition operator nor the replication operator.

Systems for the weak open congruence enjoy the tau laws in Figure 2. Let AS_o^w be the system $AS_o \cup \{T1, T2, T3, T4\}$. The proofs of the following soundness results are straightforward.

Proposition 8. (i) AS_o is sound for \sim_o^\emptyset . (ii) AS_o^w is sound for \simeq_o^\emptyset .

Most of the completeness proofs rely on some kind of normal forms. In our situation we need the following version of normal form.

M1	$\mu P = \mu' P$	if $\mu \Leftrightarrow \mu'$
M2	$[x=y]P = [x=y]P\{y/x\}$	
M3	$[x=y](P+Q) = [x=y]P+[x=y]Q$	
S1	$P+\mathbf{0} = P$	
S2	$P+Q = Q+P$	
S3	$P+(Q+R) = (P+Q)+R$	
S4	$[x=y]P+P = P$	
L1	$(x)\mathbf{0} = \mathbf{0}$	
L2	$(x)(y)P = (y)(x)P$	
L3	$(x)(P+Q) = (x)P+(x)Q$	
L4	$(x)\pi.P = \pi.(x)P$	if $x \notin n(\pi)$
L5	$(x)\pi.P = \mathbf{0}$	if $x \in \text{subj}(\pi)$
L6	$(x)[y=z]P = [y=z](x)P$	if $x \notin \{y, z\}$
L7	$(x)C[[x=y]P] = (x)C[\mathbf{0}]$	x, y are not bound in $C[\]$ and $x \neq y$

Fig. 1. The Strong System AS_o

T1	$\lambda.\tau.P = \lambda.P$
T2	$P + \tau.P = \tau.P$
T3	$\lambda.(P + \tau.Q) = \lambda.(P + \tau.Q) + \lambda.Q$
T4	$\tau.P = \tau.(P + [x=y]\tau.P)$

Fig. 2. The Tau Laws

Definition 9. A process P is in strong normal form on $V \supseteq \text{fn}(P)$ if $P \equiv \sum_{i \in I} \mu_i \lambda_i . P_i$ and the following conditions hold

(i) For each $i \in I$ and each match sequence μ satisfying $\text{fn}(\mu) \subseteq V$ there exists some $j \in I$ such that $\mu_j \Leftrightarrow \mu_i \mu$, $\lambda_i = \lambda_j$ and $P_i \equiv P_j$.

(ii) If λ_i is not an input prefix then P_i is in strong normal form on V . If λ_i is an input prefix $a(x)$ then P_i is in strong normal form on $V \cup \{x\}$.

For a strong normal form process P let $d(P)$ be the maximal nested number of prefix operators of it.

Lemma 10. For each π -process P and $V \supseteq \text{fn}(P)$ there exists a process Q which is in strong normal form on V such that $AS_o \vdash Q = P$ and $d(Q) \leq d(P)$.

4 Internalizing Distinction

In the rest of this paper we show that AS_o is complete for \sim_\emptyset^\emptyset and that AS_o^w is complete for $\simeq_\emptyset^\emptyset$. If a proof for the former is available then it is routine to obtain a proof for the latter. So we will be confining our attention to the strong case. For that purpose we need the operation $(-)^D$, for a distinction D , defined as follows:

$$\begin{aligned}
\mathbf{0}^D &\stackrel{\text{def}}{=} \mathbf{0} \\
(a(x).P)^D &\stackrel{\text{def}}{=} a(x).P^D, \text{ where } x \notin n(D) \\
(\bar{a}x.P)^D &\stackrel{\text{def}}{=} \bar{a}x.P^D \\
(\tau.P)^D &\stackrel{\text{def}}{=} \tau.P^D \\
((x)P)^D &\stackrel{\text{def}}{=} (x)P^D, \text{ where } x \notin n(D) \\
(P+Q)^D &\stackrel{\text{def}}{=} P^D + Q^D \\
([x=y]P)^D &\stackrel{\text{def}}{=} [x=y](P\{y/x\})^{D\{y/x\}}, \text{ where } (x, y) \notin D \\
([x=y]P)^D &\stackrel{\text{def}}{=} \mathbf{0}, \text{ where } (x, y) \in D
\end{aligned}$$

The role of the operation $(-)^D$ is to transfer a D -equivalent pair to an \emptyset -equivalent pair. Notice that this definition can not be applied to the π -calculus with parallel combinator. But thanks to the expansion law, in axiomatization one can remove the parallel operator in favour of the choice combinator. One of

the properties of $(-)^D$ is that it should preserve observational equivalence. This property would be lost had we replaced the seventh clause of the above definition by the following:

$$([x=y]P)^D \stackrel{\text{def}}{=} [x=y]P^D, \text{ where } (x, y) \notin D$$

Suppose $D = \{(x, y), (y, x)\}$. It is easy to see that

$$[x=z]\bar{a}x.[x=y][y=z]P \sim_o^\emptyset [x=z]\bar{a}x.[y=z]P$$

but

$$([x=z]\bar{a}x.[x=y][y=z]P)^D \not\sim_o^\emptyset ([x=z]\bar{a}x.[y=z]P)^D$$

had we used the alternative definition.

Lemma 11. *Suppose σ respects D . Then $(P^D)\sigma \equiv (P\sigma)^{D\sigma}$.*

Proof. The lemma can be proved by structural induction. Suppose for instance $(x, y) \notin D$. Then

$$\begin{aligned} (([x=y]P)^D)\sigma &= ([x=y](P\{y/x\})^{D\{y/x\}})\sigma \\ &= [x\sigma=y\sigma](P\{y/x\})^{D\{y/x\}}\sigma \\ &\stackrel{I.H.}{=} [x\sigma=y\sigma](P\{y/x\}\sigma)^{D\{y/x\}\sigma} \\ &\stackrel{M2}{=} [x\sigma=y\sigma](P\{y/x\}\sigma)^{D\{y/x\}\sigma}\{y\sigma/x\sigma\} \\ &\stackrel{I.H.}{=} [x\sigma=y\sigma](P\{y/x\}\sigma\{y\sigma/x\sigma\})^{D\{y/x\}\sigma\{y\sigma/x\sigma\}} \\ &= [x\sigma=y\sigma](P\sigma\{y\sigma/x\sigma\})^{D\sigma\{y\sigma/x\sigma\}} \\ &= ([x\sigma=y\sigma]P\sigma)^{D\sigma} \\ &= (([x=y]P)\sigma)^{D\sigma} \end{aligned}$$

If $(x, y) \in D$ then $(([x=y]P)^D)\sigma \equiv \mathbf{0} \equiv (([x=y]P)\sigma)^{D\sigma}$. \square

Lemma 12. *If $P^D \xrightarrow{\lambda} P'$ then there exists some P_1 such that $P \xrightarrow{\lambda} P_1$ and $P_1^D \equiv P'$.*

Proof. We take a look at only one case. Suppose $P \equiv [x=y]P_2$. Then by definition it must be the case that $(x, y) \notin D$ and $x = y$. So $P \equiv [x=x]P_2$ and $P^D \equiv [x=x]P_2^D$. Consequently $P_2^D \xrightarrow{\lambda} P'$. By induction hypothesis $P_2 \xrightarrow{\lambda} P_1$ for some P_1 such that $P_1^D \equiv P'$. It follows that $P \xrightarrow{\lambda} P_1$. \square

Lemma 13. *If $P \xrightarrow{\lambda} P_1$ then $P^D \xrightarrow{\lambda} P_1^D$.*

Proof. Suppose P is $[x=y]P_2$. Then it must be the case that $x = y$. Therefore $P^D \equiv [x=y]P_2^D$. The rest of the argument is easy. \square

Corollary 14. *Suppose σ respects D . If $P\sigma \xrightarrow{\lambda} P'$ then $(P^D)\sigma \xrightarrow{\lambda} (P')^{D\sigma}$.*

Lemma 15. *Suppose D_1 and D_2 are distinctions. If $P^{D_1} \sim_o^{D_2} Q^{D_1}$ then $P \sim_o^{D_1 \cup D_2} Q$.*

Proof. Suppose D is a distinction. Let \mathcal{R}^D be

$$\{(P, Q) \mid P^{D_1} \sim_o^{D_2} Q^{D_1}, D_1 \text{ and } D_2 \text{ are distinctions, } D_1 \cup D_2 = D\}$$

We prove that $\mathcal{R} = \{\mathcal{R}^D\}$ is a strong open bisimulation. Suppose σ respects D and $P\sigma \xrightarrow{\lambda} P'$. Then $(P^{D_1})\sigma \xrightarrow{\lambda} (P')^{D_1\sigma}$ by Corollary 14. It follows from $P^{D_1} \sim_o^{D_2} Q^{D_1}$ and Lemma 11 that $(Q^{D_1})\sigma \equiv (Q\sigma)^{D_1\sigma} \xrightarrow{\lambda} Q_1$ for some Q_1 to match up the action $(P^{D_1})\sigma \xrightarrow{\lambda} (P'\sigma)^{D_1\sigma}$. By Lemma 12 some Q' exists such that $Q\sigma \xrightarrow{\lambda} Q'$ and $(Q')^{D_1\sigma} \equiv Q_1$. There are two cases:

- λ is not a restricted output action. Then $(P')^{D_1\sigma} \sim_o^{D_2\sigma} (Q')^{D_1\sigma}$ by the choice of Q' . Consequently $(P', Q') \in \mathcal{R}^{D\sigma}$.
- λ is a restricted output action, say $\bar{a}(x)$. Then $(P')^{D_1\sigma} \sim_o^{D'} (Q')^{D_1\sigma}$ by the choice of Q' , where D' is $D_2\sigma \cup \{x\} \times fn(P^{D_1}\sigma + Q^{D_1}\sigma) \cup fn(P^{D_1}\sigma + Q^{D_1}\sigma) \times \{x\}$. Notice that $fn(P^{D_1}\sigma + Q^{D_1}\sigma) \subseteq fn(P\sigma + Q\sigma)$. By Lemma 6 one has $(P')^{D_1\sigma} \sim_o^{D''} (Q')^{D_1\sigma}$, where D'' is $D_2\sigma \cup \{x\} \times fn(P\sigma + Q\sigma) \cup fn(P\sigma + Q\sigma) \times \{x\}$. It follows that $(P', Q') \in \mathcal{R}^{D\sigma \cup \{x\} \times fn(P\sigma + Q\sigma) \cup fn(P\sigma + Q\sigma) \times \{x\}}$.

This completes the proof. \square

Corollary 16. *If $P^D \sim_o^\emptyset Q^D$ then $P \sim_o^D Q$.*

5 A General Completeness Result

We first prove some auxiliary lemmas.

Lemma 17. *If $\mu \Leftrightarrow \mu'$ then $AS_o \vdash (\mu P)^D = (\mu' P)^D$.*

Proof. Suppose μ is $[x_1=y_1] \dots [x_n=y_n]$ and μ' is $[a_1=b_1] \dots [a_m=b_m]$. Then

$$\begin{aligned}
(\mu P)^D &= \mu(P\{y_1/x_1\} \dots \{y_n/x_n\})^{D\{y_1/x_1\} \dots \{y_n/x_n\}} \\
&= \mu\mu'(P\{y_1/x_1\} \dots \{y_n/x_n\})^{D\{y_1/x_1\} \dots \{y_n/x_n\}} \\
&= \mu\mu'(P\{y_1/x_1\} \dots \{y_n/x_n\}\{b_1/a_1\} \dots \{b_m/a_m\})^{D\{y_1/x_1\} \dots \{y_n/x_n\}\{b_1/a_1\} \dots \{b_m/a_m\}} \\
&= \mu\mu'(P\{b_1/a_1\} \dots \{b_m/a_m\})^{D\{b_1/a_1\} \dots \{b_m/a_m\}} \\
&= \mu'(P\{b_1/a_1\} \dots \{b_m/a_m\})^{D\{b_1/a_1\} \dots \{b_m/a_m\}} \\
&= (\mu' P)^D
\end{aligned}$$

The third equality holds because $\{b_1/a_1\} \dots \{b_m/a_m\}$ and $\{y_1/x_1\} \dots \{y_n/x_n\}\{b_1/a_1\} \dots \{b_m/a_m\}$ are the same map using the fact that $\mu \Leftrightarrow \mu'$. \square

Lemma 18. *Suppose μ respects D . Then $AS_o \vdash (\mu P)^D = \mu(P\sigma_\mu)^{D\sigma_\mu}$.*

Proof. Suppose σ_μ is $\{a_1/x_1^1, \dots, a_1/x_{m_1}^1, \dots, a_n/x_1^n, \dots, a_n/x_{m_n}^n\}$. Then clearly

$$\sigma_\mu = \{a_1/x_1^1\} \dots \{a_1/x_{m_1}^1\} \dots \{a_n/x_1^n\} \dots \{a_n/x_{m_n}^n\}$$

as functions. Let μ' be $[a_1 = x_1^1] \dots [a_1 = x_{m_1}^1] \dots [a_n = x_1^n] \dots [a_n = x_{m_n}^n]$. Then $\mu \Leftrightarrow \mu'$. By Lemma 17,

$$\begin{aligned}
(\mu P)^D &= (\mu' P)^D \\
&= \mu'(P\{a_1/x_1^1\} \dots \{a_1/x_{m_1}^1\} \dots \{a_n/x_1^n\} \dots \{a_n/x_{m_n}^n\})^{D\{a_1/x_1^1\} \dots \{a_1/x_{m_1}^1\} \dots \{a_n/x_1^n\} \dots \{a_n/x_{m_n}^n\}} \\
&= \mu'(P\sigma_\mu)^{D\sigma_\mu} \\
&= \mu(P\sigma_\mu)^{D\sigma_\mu}
\end{aligned}$$

We are done. \square

The next lemma describes a generalized completeness result.

Lemma 19. *If $P \sim_o^D Q$ then $AS_o \vdash P^D = Q^D$.*

Proof. The proof is carried out by induction on $d(P) + d(Q)$. We may assume that both P and Q are strong normal forms on $fn(P+Q)$. Suppose $P \equiv \sum_{i \in I} \mu_i \lambda_i \cdot P_i$ and $Q \equiv \sum_{j \in J} \mu_j \lambda_j \cdot Q_j$. Let $(\mu_i \lambda_i \cdot P_i)^D$ be a summand of P^D , meaning that $\mu_i \lambda_i \cdot P_i$ is a summand of P and μ_i respects D . Let σ be σ_{μ_i} . It is easy to see that σ respects D . It follows from $P\sigma \xrightarrow{\lambda_i \sigma} P_i \sigma$ and $P \sim_o^D Q$ that Q must contain a summand $\mu_j \lambda_j \cdot Q_j$ such that $\mu_i \Rightarrow \mu_j$, $\lambda_i \sigma \equiv \lambda_j \sigma$ and that $Q\sigma \xrightarrow{\lambda_j \sigma} Q_j \sigma$ matches up $P\sigma \xrightarrow{\lambda_i \sigma} P_i \sigma$. Since Q is in strong normal form on $fn(P+Q)$, there must be a summand $\mu_{j'} \lambda_{j'} \cdot Q_{j'}$ of Q such that $\mu_{j'} \Leftrightarrow \mu_i$.

– If $\lambda_i \sigma$ is not a restricted output action then $P_i \sigma \sim_o^{D\sigma} Q_{j'} \sigma$. It follows by induction hypothesis that $AS_o \vdash (P_i \sigma)^{D\sigma} = (Q_{j'} \sigma)^{D\sigma}$. Consequently

$$\begin{aligned}
AS_o \vdash Q^D &= Q^D + (\mu_{j'} \lambda_{j'} \cdot Q_{j'})^D \\
&= Q^D + (\mu_i \lambda_j \cdot Q_j)^D \\
&= Q^D + \mu_i (\lambda_j \sigma \cdot Q_j \sigma)^{D\sigma} \\
&= Q^D + \mu_i \lambda_j \sigma \cdot (Q_j \sigma)^{D\sigma} \\
&\stackrel{I.H.}{=} Q^D + \mu_i \lambda_i \sigma \cdot (P_i \sigma)^{D\sigma} \\
&= Q^D + (\mu_i \lambda_i \cdot P_i)^D
\end{aligned}$$

where the second equality holds by Lemma 17 and the third equality holds by Lemma 18.

- If $\lambda_i\sigma$ is a restricted output action $\bar{a}(x)$ then $P_i\sigma \sim_o^{D'} Q_j\sigma$ where D' is $D\sigma \cup \{x\} \times fn(P\sigma+Q\sigma) \cup fn(P\sigma+Q\sigma) \times \{x\}$. By induction hypothesis one has that $AS_o \vdash (P_i\sigma)^{D'} = (Q_j\sigma)^{D'}$. Suppose we can prove

$$AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).(Q_j\sigma)^{D\sigma} \quad (3)$$

Then

$$\begin{aligned} AS_o \vdash Q^D &= Q^D + (\mu_i \bar{a}(x).Q_j)^D \\ &= Q^D + \mu_i \bar{a}\sigma(x).(Q_j\sigma)^{D\sigma} \\ &\stackrel{(3)}{=} Q^D + \mu_i \bar{a}\sigma(x).(P_i\sigma)^{D\sigma} \\ &= Q^D + (\mu_i \bar{a}(x).P_i)^D \end{aligned}$$

It follows that $AS_o \vdash Q^D = Q^D + P^D$. Similarly $AS_o \vdash P^D = P^D + Q^D$. Hence $AS_o \vdash P^D = Q^D$.

It remains to establish (3). First we show that $AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).(P_i\sigma)^{D'}$. There are two cases:

- Suppose there is no context $C[\]$ such that $P_i\sigma \equiv C[[x=y]R]$ or $P_i\sigma \equiv C[[y=x]R]$ for a free name y in $fn(P\sigma+Q\sigma)$ and a process R . Then it is easy to see from the definition of $(\cdot)^D$ that $(P_i\sigma)^{D'} \equiv (P_i\sigma)^{D\sigma}$. Consequently $AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).(P_i\sigma)^{D'}$.
- If there is a context $C[\]$ such that $P_i\sigma \equiv C[[x=y]R]$ for some free name y in $fn(P\sigma+Q\sigma)$ and some process R . Then by definition one has that

$$(P_i\sigma)^{D'} \equiv (C[\mathbf{0}])^{D'}$$

and by L8a that

$$AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).(C[\mathbf{0}])^{D\sigma}$$

Repeat the above construction until we get some P_1 such that

$$(P_i\sigma)^{D'} \equiv P_1^{D'}$$

and

$$AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).P_1^{D\sigma}$$

Now the previous case applies to P_1 , meaning that

$$AS_o \vdash \bar{a}(x).P_1^{D\sigma} = \bar{a}(x).P_1^{D'}$$

It follows that

$$AS_o \vdash \bar{a}(x).(P_i\sigma)^{D\sigma} = \bar{a}(x).(P_i\sigma)^{D'}$$

Similarly $AS_o \vdash \bar{a}(x).(Q_j\sigma)^{D\sigma} = \bar{a}(x).(Q_j\sigma)^{D'}$. Then (3) follows from $AS_o \vdash (P_i\sigma)^{D'} = (Q_j\sigma)^{D'}$. \square

Corollary 20. *For the finite π -processes without the mismatch combinator the following equivalences hold: $P^D \sim_o^\emptyset Q^D$ if and only if $P \sim_o^D Q$ if and only if $AS_o \vdash P^D = Q^D$.*

Proof. By Corollary 16, $P^D \sim_o^\emptyset Q^D$ implies $P \sim_o^D Q$. By Lemma 19, $P \sim_o^D Q$ implies $AS_o \vdash P^D = Q^D$. By Proposition 8, $AS_o \vdash P^D = Q^D$ implies $P^D \sim_o^\emptyset Q^D$. \square

Theorem 21. *AS_o is complete for \sim_o^\emptyset on the π -processes without the mismatch combinator.*

Proof. Take D in the Corollary 20 to be the empty distinction. \square

Using the technique that should be familiar to the reader by now, one can generalize the above arguments to the weak case. We state the result without any proof.

Lemma 22. *If $P \approx_o^D Q$ then $AS_o^w \vdash P^D = Q^D$.*

Theorem 23. *AS_o^w is complete for \simeq_o^\emptyset on the π -processes without the mismatch combinator.*

6 Comment

The short paper focuses on a particular axiom for the localization operator in π -calculus. This problem has been looked at by Lin in his symbolic approach to axiomatization ([13]) and by Parrow and Sangiorgi ([16]). In the symbolic approach equality reasoning is done under the assumption of a set of equational and/or inequational conditions on names. Under a complete condition the match and mismatch operators concerning a localized name in a process can be removed. This achieves the same effect as the law L7. Lin's method is applicable to π -calculus both with the mismatch combinators and without the mismatch combinators. It has to be said however that even if the π -calculus is without the mismatch operator the symbolic system has to deal with inequational conditions. Lin has studied the early/late strong/weak congruences, while Li has investigated the strong/weak open congruences ([12]). Parrow and Sangiorgi have worked on the problem in a non-symbolic framework. The π -calculus they used is the one with the mismatch operator. The underlying idea is similar to that of Lin's using the fact that the mismatch operator allows to internalize conditions to a great extent. The contribution of this short paper is to show how one can deal with the axiomatization of the π -calculus without the mismatch operator in a non-symbolic fashion.

Acknowledgement The author would like to thank Xiaoju Dong and Zhenrong Yang for helpful discussions.

References

1. Dong X, Fu Y. Observing Asymmetry and Mismatch. APLAS'03, Lecture Notes in Computer Science 2895, Springer, 2003, 2~19.
2. Fu Y. A Proof Theoretical Approach to Communications, ICALP'97, Lecture Notes in Computer Science 1256, Springer, 1997, 325~335.
3. Fu Y. Bisimulation Lattice of Chi Processes, ASIAN'98, Lecture Notes in Computer Science 1538, Springer, 1998, 245~262.
4. Fu Y. Variations on Mobile Processes, Theoretical Computer Science, 1999, 221(1-2): 327~368.
5. Fu Y. Open Bisimulations of Chi Processes. CONCUR'99, Lecture Notes in Computer Science 1664, Springer, 1999, 304~319.
6. Fu Y, Yang Z. Chi Calculus with mismatch. CONCUR 2000, Lecture Notes in Computer Science 1877, Springer, 2000, 596~610.
7. Fu Y, Yang Z. The Ground Congruence for Chi Calculus. FST&TCS 2000, Lecture Notes in Computer Science 1974, Springer, 2000, 385~396.
8. Fu Y. Bisimulation Congruences of Chi Calculus. Information and Computation, 2003, 184(1): 201~226.
9. Fu Y, Yang Z. Understanding the Mismatch Combinator in Chi Calculus. Theoretical Computer Science, 2003, 290(1): 779~830.
10. Fu Y, Yang Z. Tau Laws for Pi Calculus. Theoretical Computer Science, 2003, 308(1-3): 55~130.
11. Hennessy M, Lin H. Symbolic Bisimulations. Theoretical Computer Science, 1995, 138(2): 353~369.
12. Li Z. Theories and Algorithms for the Verification of Bisimulation Equivalences in Value-Passing CCS and the π -Calculus. PhD Thesis (Changsha Institute of Technology), 1999.
13. Lin H. Complete Inference Systems for Weak Bisimulation Equivalences in the π -Calculus. Proceedings of Sixth International Joint Conference on the Theory and Practice of Software Development, Lecture Notes in Computer Science 915, Springer, 1995, 187~201.
14. Milner R. Communication and Concurrency. Prentice Hall, 1989.
15. Milner R, Parrow J, Walker D. A Calculus of Mobile Processes. Information and Computation, 1992, 100(1): 1~40 (Part I), 41~77 (Part II).
16. Parrow J, Sangiorgi S. Algebraic Theories for Name-Passing Calculi. Information and Computation, Academic Press, 1995, 120(2): 174~197.
17. Parrow J, Victor B. The Fusion Calculus: Expressiveness and Symmetry in Mobile Processes. LICS'98, IEEE Computer Society, 1998, 176~185.
18. Sangiorgi D. A Theory of Bisimulation for π -Calculus. Acta Informatica, 1996, 3(1): 69~97.