

A Case Study on Evaluating Encodings Between Process Calculi*

Christopher Lippert^c, Stephan Mennicke^c, Rob J. van Glabbeek^{a,b}, Ursula Goltz^c

^a*Data61, CSIRO, Sydney, Australia*

^b*School of Computer Science and Engineering, University of New South Wales, Sydney, Australia*

^c*Institute for Programming and Reactive Systems, TU Braunschweig, Germany*

Abstract

Two process calculi may be compared in their expressive power using encodings between them. Not every such translation is meaningful and therefore quality criteria are necessary. There are several frameworks to assess the quality of encodings. A rather prominent approach consists of five criteria for valid encodings collected by Gorla. An alternative notion of validity up to a semantic equivalence was proposed by one of the authors. Here the quality of the encoding is measured by the strength of the semantic equivalence that makes it valid.

In this paper we analyse two well-known translations from the synchronous into the asynchronous π -calculus, both meeting Gorla's criteria, and evaluate the encodings' validity up to different semantic equivalences.

While Honda and Tokoro's translation is valid only for the weakest of the considered behavioural equivalences, Boudol's encoding also respects stronger ones. One could hence argue that Boudol's translation is more faithful.

Keywords: process calculi, quality criteria for encodings, π -calculus

1. Introduction

Since the late 1970s, a large number of process calculi has been invented and studied, see e.g. [2]. The need for new process calculi emerged with different requirements for analysability and applicability. Due to the significant number of process calculi, techniques to bring order into the process calculi landscape are highly useful. One of those techniques consists of the comparison of the expressive power of process languages. A well-established way of doing this is based on the existence of convincing translations between them.

Translations between process calculi help us understand their differences and commonalities. A translation may also be useful to make existing tools for the target language available for the source language.

A calculus \mathcal{B} is at least as expressive as \mathcal{A} if there exists a translation, mapping every \mathcal{A} expression to an expression of \mathcal{B} such that certain criteria are met. The criteria are necessary, because not every translation is suitable or meaningful. The literature provides a wide variety of translations between process calculi for which the authors often establish their own criteria satisfied by their translation, cf. [6]. Unfortunately, due to the very different criteria, it is hard to compare two translations with each other. To obtain formal arguments comparing two translations, a unified set of quality criteria is required. Gorla proposes a set of five criteria to assess process calculi encodings¹ [6].

A different approach for assessing the quality of encodings, also applicable to languages other than process calculi, is proposed by one of the authors in [4]. There, the concept of a *valid* encoding up to some semantic equivalence \sim is defined, demanding that the meaning of an expression and its translation are equivalent. This criterion has already been used to compare Milner's CCS with Hoare's CSP [4].

In this paper we analyse the validity, in the sense of [4], of two translations from a synchronous into an asynchronous variant of the π -calculus. Both are known to satisfy Gorla's criteria [5]. One encoding is given by Boudol [3], the other by Honda and Tokoro [7]. Although source and target language are very similar, the encodings use different ideas to encode synchrony by asynchronous primitives. Hence, the encodings serve as perfect candidates for a case study on the concept of validity introduced in [4]. We find an answer to the questions to what extent the validity of both encodings depends on the choice of the equivalence, and if there is any difference in the translations' quality. In Gorla's framework both translations have equally good properties, i.e. all five of Gorla's criteria are met [5].

We choose several semantic equivalences under which we evaluate the validity of the encodings. For each translation we find equivalences up to which they are valid and some for which we disprove validity. Moreover, we find a potential difference in the quality of the translations by presenting equivalences for which Boudol's encoding is valid, but Honda and Tokoro's encoding is not. Conversely, we conjecture that there is no natural equivalence for which Honda and Tokoro's encoding is valid, but Boudol's is not.

*This work was partially supported by the DFG (German Research Foundation).

¹We use the terms *encodings* and *translations* synonymously.

In Sect. 2, we give preliminary definitions. Sect. 3 defines syntax and semantics of our variants of the synchronous and asynchronous π -calculus. Furthermore, we introduce the equivalence notions employed in this paper. In Sect. 4, we present Boudol’s translation and discuss up to which equivalences it is valid. Analogously, we discuss Honda and Tokoro’s encoding in Sect. 5. Sect. 6 concludes.

2. Valid Translations between Languages

In this section, we introduce the notions of language, translation and validity of a translation according to [4]. The definitions may be applied to arbitrary translations between formal languages. However, the focus of this paper is on process calculi.

Definition 1. Let $\mathcal{D}_{\mathcal{L}}$ be a set. A *language* is a pair $\mathcal{L} = (\mathbb{T}_{\mathcal{L}}, \llbracket \cdot \rrbracket_{\mathcal{L}})$ where $\mathbb{T}_{\mathcal{L}}$ is the set of all valid expressions and $\llbracket \cdot \rrbracket_{\mathcal{L}} : \mathbb{T}_{\mathcal{L}} \rightarrow \mathcal{D}_{\mathcal{L}}$ is a function which associates with each expression $P \in \mathbb{T}_{\mathcal{L}}$ its meaning $\llbracket P \rrbracket_{\mathcal{L}} \in \mathcal{D}_{\mathcal{L}}$.

$\mathcal{D}_{\mathcal{L}}$ is called the *semantic domain of* $(\mathbb{T}_{\mathcal{L}}, \llbracket \cdot \rrbracket_{\mathcal{L}})$. If $\mathcal{D}_{\mathcal{L}}$ is the set of all labelled transition systems, then $\llbracket \cdot \rrbracket_{\mathcal{L}}$ may be the function mapping each \mathcal{L} -term $T \in \mathbb{T}_{\mathcal{L}}$ to its transition system $\llbracket T \rrbracket_{\mathcal{L}}$ given by some structural operational semantics.

In [4] languages \mathcal{L} with process variables are considered, drawn from a set \mathcal{X} , and accordingly $\mathcal{D}_{\mathcal{L}}$ is of type $(\mathcal{X} \rightarrow \mathbf{V}) \rightarrow \mathbf{V}$ for some set of values \mathbf{V} . Here an element of $\mathcal{X} \rightarrow \mathbf{V}$ is a *valuation*, associating a value from \mathbf{V} with each process variable, and the meaning of an expression is a value that depends on the choice of such a valuation. Since in this paper we deal with languages without process variables, we can instead take $\mathcal{D}_{\mathcal{L}}$ to be \mathbf{V} .

Definition 2. Let $\mathcal{L}_1, \mathcal{L}_2$ be languages. A *translation from \mathcal{L}_1 to \mathcal{L}_2* is a function $\mathcal{T} : \mathbb{T}_{\mathcal{L}_1} \rightarrow \mathbb{T}_{\mathcal{L}_2}$.

We call \mathcal{L}_1 the *source language* and \mathcal{L}_2 the *target language* of \mathcal{T} . We require the behaviour encoded by a source term $P \in \mathbb{T}_{\mathcal{L}_1}$ to be preserved by its translation $\mathcal{T}(P)$.

Definition 3. Let \mathcal{L}_i ($i = 1, 2$) be languages with semantic domains $\mathcal{D}_{\mathcal{L}_i}$ and let \sim be an equivalence on $\mathcal{D}_{\mathcal{L}_1} \cup \mathcal{D}_{\mathcal{L}_2}$. A translation $\mathcal{T} : \mathbb{T}_{\mathcal{L}_1} \rightarrow \mathbb{T}_{\mathcal{L}_2}$ is *valid up to \sim* iff \mathcal{T} is compositional and $\llbracket \mathcal{T}(P) \rrbracket_{\mathcal{L}_2} \sim \llbracket P \rrbracket_{\mathcal{L}_1}$ for all $P \in \mathbb{T}_{\mathcal{L}_1}$.

Here a translation \mathcal{T} is called *compositional* if for every n -ary operator f in the source language one has

$$\mathcal{T}(f(P_1, \dots, P_n)) \stackrel{\alpha}{\equiv} C_f(\mathcal{T}(P_1), \dots, \mathcal{T}(P_n))$$

for some n -hole context C_f . Here $\stackrel{\alpha}{\equiv}$ denotes equivalence up to α -conversion, i.e., renaming of bound names. The definition in [4] is more general, because it also covers language constructs such as recursion; this generality is not needed here. In [4] the use of the above definition is restricted to languages in which all semantic values are denotable by closed terms; this condition is trivially fulfilled for the

languages considered here. In fact, formally the semantic domain $\mathcal{D}_{\mathcal{L}}$ of the languages \mathcal{L} occurring in this paper consists of the (closed) terms of \mathcal{L} up to α -conversion—the real semantics is given by an appropriate equivalence relation on these expressions. In addition, since in this paper we investigate languages that make no use of process variables, the translations under investigation are automatically what is called “fv-translations” in [4].

In [4], besides the notion of a valid translation, also the concept of a *correct* translation up to an equivalence \sim is introduced. This definition requires \sim to be a congruence for the source language. As this condition is not met by the semantic equivalences of this paper, we will not apply the notion of a correct translation here.

3. Process Calculi

In this section we introduce two versions of the π -calculus, namely a subset π_{ms} of the (synchronous) π -calculus and an asynchronous π -calculus π_{ma} . We specify syntax and semantics, and we list semantic equivalences under which we will compare processes of the source language π_{ms} with translated processes of the target language π_{ma} .

3.1. A Synchronous π -Calculus

The (synchronous) π -calculus was first introduced by Milner, Parrow and Walker [10]. Here we use a subset of their operators that was first employed in [9].

Definition 4. Let \mathcal{N} be a set of *names*. The syntax of the *synchronous mini- π -calculus*, π_{ms} , is defined by the grammar

$$P ::= \mathbf{0} \mid \bar{x}z.P \mid x(y).P \mid P|P \mid (y)P \mid !P$$

where $x, y, z \in \mathcal{N}$.

$\mathbf{0}$ denotes the empty process. $\bar{x}z$ stands for an output guard that sends the name z along the channel x . $x(y)$ denotes an input guard that waits for a name to be transmitted along the channel named x . Upon receipt, the name is substituted for y in the subsequent process. $P|Q$ ($P, Q \in \pi_{\text{ms}}$) denotes a parallel composition between P and Q . $!P$ is the replication construct and $(y)P$ restricts the scope of name y to P .

The input construct $x(y).P$ binds the name y in P ; similarly the restriction $(y)Q$ binds y in Q . We denote the bound names of P by $\text{bn}(P)$, free names by $\text{fn}(P)$, and define $\text{n}(P) := \text{bn}(P) \cup \text{fn}(P)$. With $P\{w/y\}$ we denote the result of substituting w for all free occurrences of y in P , with change of bound names to avoid name captures.

Following [11, 9], we give two semantics of π_{ms} : a labelled transition semantics and a reduction semantics. The labels of the former are drawn from a set of actions $\text{Act} := \{\bar{x}y, x(y), \bar{x}(y) \mid x, y \in \mathcal{N}\} \cup \{\tau\}$. We define free and bound names on transition labels:

$$\begin{array}{ll} \text{fn}(\tau) = \emptyset & \text{bn}(\tau) = \emptyset \\ \text{fn}(\bar{x}z) = \{x, z\} & \text{bn}(\bar{x}z) = \emptyset \\ \text{fn}(x(y)) = \{x\} & \text{bn}(x(y)) = \{y\} \\ \text{fn}(\bar{x}(y)) = \{x\} & \text{bn}(\bar{x}(y)) = \{y\} \end{array}$$

For $\alpha \in \text{Act}$ we define $\text{n}(\alpha) := \text{bn}(\alpha) \cup \text{fn}(\alpha)$.

Definition 5. The *labelled transition relation* of π_{ms} is the smallest relation $\longrightarrow \subseteq \mathbb{T}_{\pi_{\text{ms}}} \times \text{Act} \times \mathbb{T}_{\pi_{\text{ms}}}$, satisfying the rules of Fig. 1.

$$\begin{array}{c}
\text{(OUTPUT-ACT)} \frac{}{\bar{x}z.P \xrightarrow{\bar{x}z} P} \\
\text{(INPUT-ACT)} \frac{w \notin \text{fn}((y)P)}{x(y).P \xrightarrow{x(w)} P\{w/y\}} \\
\text{(PAR)} \frac{P \xrightarrow{\alpha} P' \quad \text{bn}(a) \cap \text{fn}(Q) = \emptyset}{P|Q \xrightarrow{\alpha} P'|Q} \\
\text{(COM)} \frac{P \xrightarrow{\bar{x}z} P' \quad Q \xrightarrow{x(y)} Q'}{P|Q \xrightarrow{\tau} P'|Q'\{z/y\}} \\
\text{(CLOSE)} \frac{P \xrightarrow{\bar{x}(w)} P' \quad Q \xrightarrow{x(w)} Q'}{P|Q \xrightarrow{\tau} (w)(P'|Q')} \\
\text{(RES)} \frac{P \xrightarrow{\alpha} P' \quad y \notin \text{n}(a)}{(y)P \xrightarrow{\alpha} (y)P'} \\
\text{(OPEN)} \frac{P \xrightarrow{\bar{x}y} P' \quad y \neq x \quad w \notin \text{fn}((y)P')}{(y)P \xrightarrow{\bar{x}(w)} P'\{w/y\}} \\
\text{(REP-ACT)} \frac{P \xrightarrow{\alpha} P'}{!P \xrightarrow{\alpha} P'!P} \\
\text{(REP-COMM)} \frac{P \xrightarrow{\bar{x}z} P' \quad P \xrightarrow{x(y)} P''}{!P \xrightarrow{\tau} (P'|P''\{z/y\})!P} \\
\text{(REP-CLOSE)} \frac{P \xrightarrow{\bar{x}(w)} P' \quad P \xrightarrow{x(w)} P''}{!P \xrightarrow{\tau} ((w)(P'|P''))!P}
\end{array}$$

Figure 1: SOS rules for the synchronous mini- π -calculus. PAR, COM and CLOSE also have symmetric rules.

Another common way to describe the behaviour of a process is a reduction semantics [9]. Instead of taking each label into account, here we are only interested in the actual communication steps a π_{ms} process may perform. A crucial part of this type of semantics is handled by *structural congruence* \equiv_{R} , the smallest congruence relation on $\mathbb{T}_{\pi_{\text{ms}}}$ satisfying the following five rules:

1. Processes that only differ in the choice of bound names are identified; this is known as α -conversion.
2. $(\mathbb{T}_{\pi_{\text{ms}}}/\equiv_{\text{R}}, |, \mathbf{0})$ is an Abelian monoid, i. e. (i) $\mathbf{0}|P \equiv_{\text{R}} P$, (ii) $(P|Q)|R \equiv_{\text{R}} P|(Q|R)$ and (iii) $P|Q \equiv_{\text{R}} Q|P$, for all $P, Q, R \in \mathbb{T}_{\pi}$.
3. $!P \equiv_{\text{R}} P|!P$.
4. $(x)\mathbf{0} \equiv_{\text{R}} \mathbf{0}$, $(x)(y)P \equiv_{\text{R}} (y)(x)P$.
5. If $x \notin \text{fn}(P)$, then $(x)(P|Q) \equiv_{\text{R}} P|(x)Q$.

Definition 6. The *reduction relation* $\longmapsto \subseteq \mathbb{T}_{\pi_{\text{ms}}} \times \mathbb{T}_{\pi_{\text{ms}}}$ is the smallest relation satisfying the rules of Fig. 2.

$$\begin{array}{c}
\text{(COM)} \frac{}{\bar{x}z.P|x(y).Q \longmapsto P|Q\{z/y\}} \\
\text{(PAR)} \frac{P \longmapsto P'}{P|Q \longmapsto P'|Q} \quad \text{(RES)} \frac{P \longmapsto P'}{(y)P \longmapsto (y)P'} \\
\text{(STRUCT)} \frac{Q \equiv_{\text{R}} P \quad P \longmapsto P' \quad P' \equiv_{\text{R}} Q'}{Q \longmapsto Q'}
\end{array}$$

Figure 2: Reduction semantics of the mini- π -calculus.

The following results show (1) that the labelled transition relations are invariant under structural congruence (\equiv_{R}), and (2) that the closure under structural congruence of the labelled transition relation restricted to τ -steps coincides with the reduction relation — (2) stems from Milner [9].

Lemma 1. (Harmony Lemma [14, Lemma 1.4.15])

1. If $P \xrightarrow{\alpha} P'$ and $P \equiv_{\text{R}} Q$ then $\exists Q'. Q \xrightarrow{\alpha} Q' \equiv_{\text{R}} P'$
2. $P \longmapsto P'$ iff $\exists P''. P \xrightarrow{\tau} P'' \equiv_{\text{R}} P'$.

3.2. The Asynchronous π -Calculus

A characteristic of synchronous communication, as used in π_{ms} , is that sending a message synchronises with receiving it, so that a process sending a message can only proceed after another party has received it. In the asynchronous π -calculus this feature is dropped, as it is not possible to specify any behaviour scheduled after a send action.

Definition 7 ([3]). The *asynchronous mini- π -calculus*, π_{ma} , consists of all processes in π_{ms} such that in each sub-term of the form $\bar{x}y.P$, $P = \mathbf{0}$.

Instead of $\bar{x}y.\mathbf{0}$, we just write $\bar{x}y$. Since the asynchronous mini- π -calculus is a subset of the synchronous one, we simply reuse the semantics defined above. π_{ma} is closed under structural congruence (\equiv_{R}), labelled transitions, as well as reductions.

3.3. Equivalences

Here we present six behavioural equivalences for π_{ms} and π_{ma} . We start with the standard notion of *early weak bisimulation* for labelled transition systems [14].

Definition 8. A symmetric binary relation \mathcal{R} on π -processes P, Q is a *early weak bisimulation* iff $P \mathcal{R} Q$ implies

1. if $P \xrightarrow{\tau} P'$ then a Q' exists with $Q \xrightarrow{\tau} Q'$ and $P' \mathcal{R} Q'$,
2. if $P \xrightarrow{\alpha} P'$ where $\alpha = \bar{x}z$ or $\bar{x}(y)$ with $y \notin \text{n}(P) \cup \text{n}(Q)$ then a Q' exists with $Q \xrightarrow{\tau} Q' \xrightarrow{\alpha} P'$ and $P' \mathcal{R} Q'$,
3. if $P \xrightarrow{x(y)} P'$ with $y \notin \text{n}(P) \cup \text{n}(Q)$ then for all w a Q' exists with $Q \xrightarrow{\tau} Q' \xrightarrow{x(y)} P'$ and $P'\{w/y\} \mathcal{R} Q'\{w/y\}$.

We denote the largest early weak bisimulation by \sim_{EWB} .

Here $y \notin \text{n}(P) \cup \text{n}(Q)$ merely ensures the usage of fresh names. A *late* weak bisimulation is obtained by requiring in Clause 3 above that the choice of Q' is independent of w ; this gives rise to a slightly finer equivalence relation.

A weaker approach does not compare all the transitions with visible labels, for these are merely *potential* transitions, that can occur only in certain contexts. Instead it just compares internal transitions, together with the information whether a state has the potential to perform an input or output over a certain channel. We treat the capability to read or write over a channel as a predicate of a process. Such predicates are called *barbs* [8].

Definition 9. Let $x, z \in \mathcal{N}$. A process P has a *strong barb* on x , $P \Downarrow_x$, iff there is a P' with $P \xrightarrow{x(z)} P'$. It has a *strong barb* on \bar{x} , $P \Downarrow_{\bar{x}}$, iff there is a P' with $P \xrightarrow{\bar{x}z} P'$ or $P \xrightarrow{\bar{x}(z)} P'$. A process P has a *weak barb* on a ($a \in \{x, \bar{x} \mid x \in \mathcal{N}\}$), $P \Downarrow_a$, iff there is a P' such that $P \xrightarrow{\tau}^* P'$ and $P' \Downarrow_a$.

Combining the notion of barbs with the transfer property of classical bisimulation for internal actions only yields *weak barbed bisimulation* [8]. Here, two related processes simulate each other's internal transitions and furthermore have the same weak barbs.

Definition 10. A symmetric relation \mathcal{R} on π -terms is a *weak barbed bisimulation* iff $P \mathcal{R} Q$ implies

1. if $P \Downarrow_a$ with $a \in \{x, \bar{x} \mid x \in \mathcal{N}\}$ then $Q \Downarrow_a$ and
2. if $P \xrightarrow{\tau} P'$ then a Q' exists with $Q \xrightarrow{\tau}^* Q'$ and $P' \mathcal{R} Q'$.

The largest weak barbed bisimulation is denoted by \sim_{WBB} .

By Lemma 1 this definition can equivalently be stated with \mapsto in the role of $\xrightarrow{\tau}$.

In *asynchronous weak barbed bisimulation* [1], only the names of output channels are observed. Input barbs are ignored here, as it is assumed that an environment is able to observe output messages, but not (missing) inputs.

Definition 11. A symmetric relation S on π -terms is an *asynchronous weak barbed bisimulation* iff $P \mathcal{R} Q$ implies

1. if $P \Downarrow_{\bar{x}}$, then $Q \Downarrow_{\bar{x}}$, and
2. if $P \xrightarrow{\tau} P'$ then a Q' exists with $Q \xrightarrow{\tau}^* Q'$ and $P' \mathcal{R} Q'$.

The largest asynchronous weak barbed bisimulation is denoted by \sim_{AWBB} .

We obtain a stronger equivalence if we consider not only output channels but also the messages sent along them.

Definition 12 ([1]). A symmetric relation \mathcal{R} on π -terms is a *weak $\sigma\tau$ -bisimulation* if \mathcal{R} meets Clauses 1 and 2 (but not necessarily 3) from Definition 8. The largest weak $\sigma\tau$ -bisimulation is denoted by $\sim_{\text{W}\sigma\tau}$.

Amadio et al. strengthen this equivalence by adding a further constraint for input transitions.

Definition 13 ([1]). A relation \mathcal{R} is a *weak asynchronous bisimulation* iff \mathcal{R} is a weak $\sigma\tau$ -bisimulation such that $P \mathcal{R} Q$ and $P \xrightarrow{\tau}^* \xrightarrow{x(y)} \xrightarrow{\tau}^* P'$ implies

- either a Q' exists satisfying a condition akin to Clause 3 of Definition 8 [1], or

- a Q' exists such that $Q \xrightarrow{\tau}^* Q'$ and $P' \mathcal{R} (Q' | \bar{x}y)$.

The largest weak asynchronous bisimulation is denoted by \sim_{WAB} .

As a last equivalence we introduce an even weaker equivalence than weak barbed bisimulation, that does not distinguish between input and output channels.

Definition 14. A process P can perform an action on channel x , $P \Downarrow_x^c$, if $P \xrightarrow{\alpha} P'$, for some P' , where α has the form $\bar{x}y$, $\bar{x}(y)$ or $x(y)$. We write $P \Downarrow_x^c$ when a P' exists with $P \xrightarrow{\tau}^* P'$ and $P' \Downarrow_x^c$.

A symmetric relation \mathcal{R} on π -terms is a *weak channel bisimulation* if $P \mathcal{R} Q$ implies

1. if $P \Downarrow_x^c$ then $Q \Downarrow_x^c$ and
2. if $P \xrightarrow{\tau} P'$ then a Q' exists with $Q \xrightarrow{\tau}^* Q'$ and $P' \mathcal{R} Q'$.

The largest weak channel bisimulation is denoted \sim_{WCB} .

We thereby obtain the following hierarchy of equivalence relations on π -calculus processes (cf. Fig. 3).

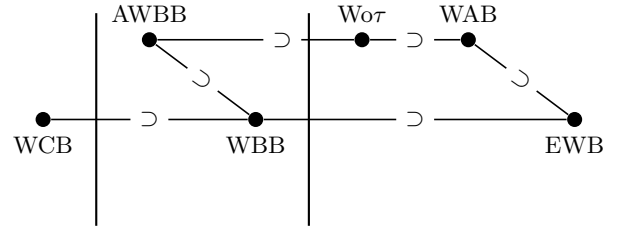


Figure 3: A hierarchy on semantic equivalence relations for π -calculus processes, with separation lines indicating where the encodings discussed in this paper pass and fail validity.

4. Boudol's Translation

In the previous section, we introduced syntax and semantics of the synchronous as well as of the asynchronous π -calculus. Furthermore, we picked several behavioural equivalences under which we want to assess two translations between these process calculi. Boudol's translation $\mathcal{T}_B : \mathbb{T}_{\pi_{\text{ms}}} \rightarrow \mathbb{T}_{\pi_{\text{ma}}}$ shall be the first [3]. The crucial task is to implement synchronous communication in the asynchronous calculus.

Suppose there is a π_{ms} -process able to perform a communication, for example $\bar{x}z.P|x(y).Q$. In the asynchronous variant of the π -calculus, there is no continuation process after an output operation. Hence, a translation into the asynchronous π -calculus has to reflect the communication on channel x as well as the guarding role of $\bar{x}z$ for P in the synchronous π -calculus. The idea of Boudol's encoding is to assign a guard to P such that this process must receive an *acknowledgement message* confirming the receipt of z . We write the sender as $P' = (\bar{x}z|u(v).P)$ where $u, v \notin \text{fn}(P)$. Symmetrically, the receiver must send the acknowledgement, i. e. $Q' = x(y).(\bar{u}v|Q)$. Unfortunately,

this simple transformation is not applicable in every case, because the protocol does not protect the channel u . u should be known to sender and receiver only, otherwise the communication may be interrupted by the environment. Therefore, we restrict the scope of u , and start by sending this private channel to the receiver. The actual message z is now send in a second stage, over a channel v , which is also made into a private channel between the two processes. The crucial observation is that in $(u)(\bar{x}u|u(v).P^*)$, the subprocess $P^* = \bar{v}z|P$ may only continue after $\bar{x}u$ was accepted by some receiver, and this receiver has acknowledged this by transmitting another channel name v on the private channel u . Boudol's translation from π_{ms} to π_{ma} is defined by the following function (where $u \neq v$):

$$\begin{aligned} \mathcal{T}_B(\mathbf{0}) &= \mathbf{0} \\ \mathcal{T}_B(\bar{x}z.P) &= (u)(\bar{x}u|u(v).(\bar{v}z|\mathcal{T}_B(P))) \quad u, v \notin \text{fn}(P) \cup \{x, z\} \\ \mathcal{T}_B(x(y).P) &= x(u).(v)(\bar{u}v|v(y).\mathcal{T}_B(P)) \quad u, v \notin \text{fn}(P) \cup \{x\} \\ \mathcal{T}_B(P|Q) &= (\mathcal{T}_B(P)|\mathcal{T}_B(Q)) \\ \mathcal{T}_B(!P) &= !\mathcal{T}_B(P) \\ \mathcal{T}_B((x)P) &= (x)\mathcal{T}_B(P) \end{aligned}$$

In the original definition, Boudol observed that $\mathbf{0}$ could be dropped from the syntax of π_{ma} in favour of the constant $\bar{x}z$, and accordingly used $\mathcal{T}_B(\mathbf{0}) = (x)(z)\bar{x}z$.

The following lemma guarantees that the names u and v used above do not occur in the translated terms $\mathcal{T}_B(P)$ and $\mathcal{T}_B(Q)$ either.

Lemma 2. *Let $P \in \mathbb{T}_{\pi_{\text{ms}}}$. Then $\text{fn}(\mathcal{T}_B(P)) = \text{fn}(P)$. Moreover, $\mathcal{T}_B(P)\{z/y\} = \mathcal{T}_B(P\{z/y\})$ for any $z, y \in \mathcal{N}$.*

Proof. A straightforward structural induction on P . \square

By construction, the encoding of Boudol is compositional. As it makes use of intermediate steps (namely the acknowledgement protocol), we must fail proving validity of the encoding up to semantics based on action labels, e. g. early weak bisimulation.

Theorem 1. \mathcal{T}_B is not valid up to \sim_{EWB} .

Proof. Let $P = \bar{x}z.\mathbf{0}$ and $\mathcal{T}_B(P) = (u)(\bar{x}u|u(v).(\bar{v}z|\mathbf{0}))$. We present the relevant parts of the labelled transition semantics:

$$\begin{array}{ccc} & & (u)(\bar{x}u|u(v).(\bar{v}z|\mathbf{0})) \\ & & \downarrow \bar{x}(c) \\ \bar{x}z.\mathbf{0} & & (u)(\mathbf{0}|c(v).(\bar{v}z|\mathbf{0})) \\ \downarrow \bar{x}z & & \downarrow c(d) \\ \mathbf{0} & & (u)(\mathbf{0}|(\bar{d}z|\mathbf{0})) \\ & & \downarrow \bar{d}z \\ & & (u)(\mathbf{0}|(\mathbf{0}|\mathbf{0})) \end{array}$$

Here, the translated term may perform an input transition $\xrightarrow{c(d)}$ the source term is not capable of. Hence, the processes are not equivalent up to early weak bisimulation. \square

Early weak bisimulation is too strong to prove the translation valid, as intermediate steps are taken into account by the equivalence. Weaker equivalences such as weak barbed bisimulation only compare the barbs of a process. Fortunately, Boudol's encoding keeps the original channel names of a sending or receiving process invariant. Hence, a translated term does exhibit the same barbs as the source term.

Lemma 3. *Let $P \in \mathbb{T}_{\pi_{\text{ms}}}$ and $a \in \{x, \bar{x} \mid x \in \mathcal{N}\}$. Then $P \downarrow_a$ iff $\mathcal{T}_B(P) \downarrow_a$.*

Proof. With structural induction on P .

- $\mathbf{0}$ and $\mathcal{T}_B(\mathbf{0})$ have the same strong barbs, namely none.
- $\bar{x}z.P$ and $\mathcal{T}_B(\bar{x}z.P)$ both have only the strong barb \bar{x} .
- $x(y).P$ and $\mathcal{T}_B(x(y).P)$ both have only strong barb x .
- The strong barbs of $P|Q$ are the union of the ones of P and Q . Using this, the case $P|Q$ follows by induction.
- The strong barbs of $!P$ are the ones of P . Using this, the case $!P$ follows by induction.
- The strong barbs of $(x)P$ are ones of P except x and \bar{x} . Using this, the case $(x)P$ follows by induction. \square

Before we prove validity of Boudol's translation up to weak barbed bisimulation, we further investigate the protocol steps established by Boudol's encoding. Let $P' = \bar{x}z.P$ and $Q' = x(y).Q$. Pick u, v not free in P and Q , with $u \neq v$. Write $P^* := \bar{v}z|\mathcal{T}_B(P)$ and $Q^* := v(y).\mathcal{T}_B(Q)$. Then

$$\begin{aligned} \mathcal{T}_B(P'|Q') &= (u)(\bar{x}u|u(v).P^*) \mid x(u).(v)(\bar{u}v|Q^*) \\ &\mapsto (u)(u(v).P^* \mid (v)(\bar{u}v|Q^*)) \\ &\mapsto (v)(P^*|Q^*) \\ &\mapsto \mathcal{T}_B(P)|(\mathcal{T}_B(Q)\{z/y\}). \end{aligned}$$

Here structural congruence is applied in omitting parallel components $\mathbf{0}$ and empty binders (u) and (v) . Now the crucial idea in our proof is that the last two reductions are *inert*, in that set of the potential behaviours of a process is not diminished by doing (internal) steps of this kind. The first reduction above in general is not inert, as it creates a commitment between a sender and a receiver to communicate, and this commitment goes at the expense of the potential of one of the two parties to do this communication with another partner. We employ a relation that captures these inert reductions in a context.

Definition 15 ([5]). Let \Longrightarrow be the smallest relation on $\mathbb{T}_{\pi_{\text{ma}}}$ such that

1. $(v)(\bar{v}y|P|v(z).Q) \Longrightarrow P|(Q\{y/z\})$,
2. if $P \Longrightarrow Q$ then $P|C \Longrightarrow Q|C$,
3. if $P \Longrightarrow Q$ then $(w)P \Longrightarrow (w)Q$,
4. if $P \equiv_{\text{R}} P' \Longrightarrow Q' \equiv_{\text{R}} Q$ then $P \Longrightarrow Q$,

where $v \notin \text{fn}(P) \cup \text{fn}(Q\{y/z\})$.

First of all observe that whenever two processes are related by \Longrightarrow , an actual reduction takes place.

Lemma 4 ([5]). *If $P \Longrightarrow Q$ then $P \mapsto Q$.*

The next two lemmas confirm that inert reductions do not diminish the potential behaviour of a process.

Lemma 5 ([5]). *If $P \Longrightarrow Q$ and $P \mapsto P'$ with $P' \not\equiv_{\mathbb{R}} Q$ then there is a Q' with $Q \mapsto Q'$ and $P' \Longrightarrow Q'$.*

Corollary 1. *If $P \Longrightarrow^* Q$ and $P \mapsto P'$ then either $P' \Longrightarrow^* Q$ or there is a Q' with $Q \mapsto Q'$ and $P' \Longrightarrow^* Q'$.*

Proof. By repeated application of Lemma 5. \square

Lemma 6. *If $P \Longrightarrow Q$ and $P \downarrow_a$ for $a \in \{x, \bar{x} \mid x \in \mathcal{N}\}$ then $Q \downarrow_a$.*

Proof. Let $(\tilde{w})P$ for $\tilde{w} = \{w_1, \dots, w_n\} \subseteq \mathcal{N}$ with $n \in \mathbb{N}$ denote $(w_1) \cdots (w_n)P$ for some arbitrary order of the (w_i) . Using a trivial variant of Lemma 1.2.20 in [14], there are $\tilde{w} \subseteq \mathcal{N}$, $x, y, z \in \mathcal{N}$ and $R, C \in \mathbb{T}_{\pi_{\text{ma}}}$, such that $x \in \tilde{w}$ and $P \equiv_{\mathbb{R}} (\tilde{w})((\bar{x}y|x(z).R)|C) \mapsto (\tilde{w})((\mathbf{0}|R\{y/z\})|C) \equiv_{\mathbb{R}} Q$. Since $P \downarrow_a$, it must be that $a=x$ or \bar{x} with $x \notin \tilde{w}$, and $C \downarrow_a$. Hence $Q \downarrow_a$. \square

The following lemma states, in terms of Gorla's framework, *operational completeness* [6]: if a source term is able to make a step, then its translation is able to simulate that step by protocol steps.

Lemma 7 ([5]). *Let $P, P' \in \mathbb{T}_{\pi_{\text{ms}}}$. If $P \mapsto P'$ then $\mathcal{T}_{\mathbb{B}}(P) \mapsto^* \mathcal{T}_{\mathbb{B}}(P')$.*

Finally, the next lemma was a crucial step in establishing *operational soundness* [6].

Lemma 8 ([5]). *Let $P \in \mathbb{T}_{\pi_{\text{ms}}}$ and $Q \in \mathbb{T}_{\pi_{\text{ma}}}$. If $\mathcal{T}_{\mathbb{B}}(P) \mapsto Q$ then there is a P' with $P \mapsto P'$ and $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P')$.*

Using these lemmas, we prove the validity of Boudol's encoding up to weak reduction bisimulation.

Theorem 2. *Boudol's encoding is valid up to \sim_{WBB} .*

Proof. Define the relation \mathcal{R} by $P \mathcal{R} Q$ iff $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$. It suffices to show that the symmetric closure of \mathcal{R} is a weak barbed bisimulation.

To show that \mathcal{R} satisfies Clause 1 of Def. 10, suppose $P \mathcal{R} Q$ and $P \downarrow_a$ for $a \in \{x, \bar{x} \mid x \in \mathcal{N}\}$. Then $\mathcal{T}_{\mathbb{B}}(P) \downarrow_a$ by Lemma 3. Since $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$, we obtain $Q \mapsto^* \mathcal{T}_{\mathbb{B}}(P)$ by Lemma 4, and thus $Q \downarrow_a$.

To show that \mathcal{R} also satisfies Clause 2, suppose $P \mathcal{R} Q$ and $P \mapsto P'$. Since $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$, by Lemmas 4 and 7 we have $Q \mapsto^* \mathcal{T}_{\mathbb{B}}(P) \mapsto^* \mathcal{T}_{\mathbb{B}}(P')$, and also $P' \mathcal{R} \mathcal{T}_{\mathbb{B}}(P')$.

To show that \mathcal{R}^{-1} satisfies Clause 1, suppose $P \mathcal{R} Q$ and $Q \downarrow_a$. Since $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$, Lemma 6 yields $\mathcal{T}_{\mathbb{B}}(P) \downarrow_a$, and Lemma 3 gives $P \downarrow_a$, which implies $P \downarrow_a$.

To show that \mathcal{R}^{-1} satisfies Clause 2, suppose $P \mathcal{R} Q$ and $Q \mapsto Q'$. Since $Q \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$, by Corollary 1 either $Q' \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P)$ or there is a Q'' with $\mathcal{T}_{\mathbb{B}}(P) \mapsto Q''$ and

$Q' \Longrightarrow^* Q''$. In the first case $P \mathcal{R} Q'$, so taking $P' := P$ we are done. In the second case, by Lemma 8 there is a P' with $P \mapsto P'$ and $Q'' \Longrightarrow^* \mathcal{T}_{\mathbb{B}}(P')$. We thus have $P' \mathcal{R} Q'$. \square

Since \sim_{AWBB} and \sim_{WCB} are weaker equivalences than \sim_{WBB} , we obtain the following:

Corollary 2. *Boudol's translation is valid up to \sim_{AWBB} and up to \sim_{WCB} .*

We now know that Boudol's translation is valid up to \sim_{AWBB} , but not up to \sim_{EWB} . A natural step is to narrow down this gap by considering equivalences in between.

Theorem 3. *Boudol's translation $\mathcal{T}_{\mathbb{B}} : \mathbb{T}_{\pi_{\text{ms}}} \rightarrow \mathbb{T}_{\pi_{\text{ma}}}$ is not valid up to \sim_{WOT} , and not up to \sim_{WAB} .*

Proof. Consider the proof of Theorem 1. $\bar{x}z.\mathbf{0}$ sends a free name along x while $(u)(\bar{x}u|u(v).(\bar{v}z|\mathbf{0}))$ sends a bound name along the same channel. Since \sim_{WOT} differentiates between free and bound names, the transition systems of $\bar{x}z$ and its translation are not \sim_{WOT} -equivalent. \square

5. Honda and Tokoro's Translation

Honda and Tokoro translate between the same languages as Boudol does [7]. Unlike Boudol's translation, communication takes place directly after synchronising along the private channel u . The synchronisation occurs in the reverse direction, because sending and receiving messages alternate, meaning that the sending process $\bar{x}z.Q$ is translated into a process that receives a message on channel x and the receiving process $x(y).R$ is translated into a process passing a message on x .

$$\begin{aligned} \mathcal{T}_{\text{HT}}(\mathbf{0}) &= \mathbf{0} \\ \mathcal{T}_{\text{HT}}(\bar{x}z.P) &= x(u).(\bar{u}z|\mathcal{T}_{\text{HT}}(P)) & u \notin \text{fn}(P) \cup \{x, z\} \\ \mathcal{T}_{\text{HT}}(x(y).P) &= (u)(\bar{x}u|u(y).\mathcal{T}_{\text{HT}}(P)) & u \notin \text{fn}(P) \cup \{x\} \\ \mathcal{T}_{\text{HT}}(P|Q) &= (\mathcal{T}_{\text{HT}}(P)|\mathcal{T}_{\text{HT}}(Q)) \\ \mathcal{T}_{\text{HT}}(!P) &= !\mathcal{T}_{\text{HT}}(P) \\ \mathcal{T}_{\text{HT}}((x)P) &= (x)\mathcal{T}_{\text{HT}}(P) \end{aligned}$$

Again, this translation is compositional. As in the case of Boudol's translation, Honda and Tokoro's translation is not valid up to equivalences based on action labels:

Theorem 4. *Honda and Tokoro's translation \mathcal{T}_{HT} is not valid up to \sim_{WOT} , \sim_{WAB} or \sim_{EWB} .*

Proof. By the counterexample $\bar{x}z.\mathbf{0}$, as for Theorem 1. \square

So again we have to use variants of barbed bisimilarity. However, the choices of \sim_{AWBB} or \sim_{WBB} , that worked for Boudol's translation, bear no fruit here.

Theorem 5. *Honda and Tokoro's translation \mathcal{T}_{HT} is not valid up to \sim_{AWBB} , and thus not up to \sim_{WBB} .*

Proof. Let $P = \bar{x}z.0$. Then $P \downarrow_{\bar{x}}$. The translation is $\mathcal{T}_{\text{HT}}(P) = x(u).(uz|0)$ and $\mathcal{T}_{\text{HT}}(P) \not\Downarrow_{\bar{x}}$. \square

One equivalence remains, namely weak channel bisimulation which does not distinguish between input and output channels. Actually, we introduced \sim_{WCB} as a rather weak equivalence after we had proven that Honda and Tokoro's encoding is not valid up to one of the original equivalences.

Theorem 6. *Honda and Tokoro's translation \mathcal{T}_{HT} is valid up to \sim_{WCB} .*

The proof is similar to the one of Theorem 2. Here we use that Lemmas 7 and 8 also apply to \mathcal{T}_{HT} [5] and Lemma 3 now holds with \downarrow_x^c in the role of \downarrow_a .

6. Conclusion

We evaluated the concept of a valid translation between two languages, introduced by one of the authors [4]. The two translations we examined were developed by Boudol and by Honda and Tokoro. Both translate from the synchronous into the asynchronous mini- π -calculus. Honda and Tokoro translate an output action into a construct that first performs an input action on the same channel and vice versa, while Boudol's encoding leaves the roles of sender and receiver names untouched.

We instantiated the concept of validity from [4] with six behavioural equivalences. Fig. 3 summarises the results.

In order to prove the validity of Honda and Tokoro's translation, we had to use the novel weak channel bisimulation that does not distinguish between input and output channels. Boudol's encoding withstands a stronger equivalence, namely weak barbed bisimilarity. Hence, Honda and Tokoro's encoding can be regarded as weaker than the one of Boudol. Whether Boudol's encoding is to be preferred, because it meets stronger requirements/equivalences, is a decision that should be driven by the requirements of an application the encoding is used for.

We only used weak equivalences, abstracting from τ -steps. The reason is that both translations implement a protocol with additional transitions, and thus are not valid up to strong equivalences.

A translation for further study would be Nestmann's encoding of the π -calculus with separate choice into the asynchronous π -calculus [12]. One could also try to state Palamidessi's separation result [13] in the current framework. It states that the π -calculus with mixed choice is strictly more expressive than the one with separate choice. After choosing an appropriate weak equivalence, it must be proven that there is no valid encoding from π with mixed choice into the π -calculus with separate choice.

Finally, we remark that translations are only comparable if their correctness or validity is proven using the same criteria. It is not useful to establish a specific set of criteria for each translation. Consequently, the use of general approaches like the ones by Gorla or the one used here leads to comparable results on expressiveness of process calculi, positive as well as negative ones.

References

- [1] Roberto M. Amadio, Ilaria Castellani & Davide Sangiorgi (1998): *On Bisimulations for the Asynchronous pi-Calculus*. TCS 195(2), pp. 291–324, doi:10.1016/S0304-3975(97)00223-5.
- [2] Jos C. M. Baeten (2005): *A brief history of process algebra*. TCS 335(2-3), pp. 131–146, doi:10.1016/j.tcs.2004.07.036.
- [3] Gerard Boudol (1992): *Asynchrony and the pi-Calculus*. Technical Report 1702, INRIA. Note.
- [4] Rob J. van Glabbeek (2012): *Musings on Encodings and Expressiveness*. In Bas Luttik & Michel A. Reniers, editors: Proc. EXPRESS/SOS'12, EPTCS 89, Open Publishing Association, pp. 81–98, doi:10.4204/EPTCS.89.7.
- [5] Rob J. van Glabbeek (2018): *On the Validity of Encodings of the Synchronous in the Asynchronous pi-calculus*. Available at <http://theory.stanford.edu/~rvg/abstracts.html#128>.
- [6] Daniele Gorla (2010): *Towards a unified approach to encodability and separation results for process calculi*. Inf. and Comp. 208(9), pp. 1031–1053, doi:10.1016/j.ic.2010.05.002.
- [7] Kohei Honda & Mario Tokoro (1991): *An Object Calculus for Asynchronous Communication*. In Pierre America, editor: Proc. ECOOP'91, LNCS 512, Springer, pp. 133–147, doi:10.1007/BFb0057019.
- [8] Robin Milner (1991): *The Polyadic pi-Calculus: A Tutorial*. Technical Report ECS-LFCS-91-180, The University of Edinburgh. Informatics Report Series.
- [9] Robin Milner (1992): *Functions as Processes*. MFCS 2, pp. 119–141, doi:10.1017/S0960129500001407.
- [10] Robin Milner, Joachim Parrow & David Walker (1992): *A Calculus of Mobile Processes, I. Information and Computation* 100(1), pp. 1–40, doi:10.1016/0890-5401(92)90008-4.
- [11] Robin Milner, Joachim Parrow & David Walker (1992): *A Calculus of Mobile Processes, II. Information and Computation* 100(1), pp. 41–77, doi:10.1016/0890-5401(92)90009-5.
- [12] Uwe Nestmann (2000): *What is a 'Good' Encoding of Guarded Choice? Information and Computation* 156, pp. 287–319, doi:10.1006/inco.1999.2822.
- [13] Catuscia Palamidessi (2003): *Comparing The Expressive Power Of The Synchronous And Asynchronous Pi-Calculi*. MFCS 13(5), pp. 685–719, doi:10.1017/S0960129503004043.
- [14] Davide Sangiorgi & David Walker (2001): *The Pi-Calculus - a theory of mobile processes*. Cambridge University Press.