Under consideration for publication in Formal Aspects of Computing

# Characterisations of Testing Preorders for a Finite Probabilistic $\pi$-Calculus

Yuxin Deng[1] and Alwen Tiu[2]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
and State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences
[2]Research School of Computer Science, The Australian National University

**Abstract.** We consider two characterisations of the may and must testing preorders for a probabilistic extension of the finite $\pi$-calculus: one based on notions of probabilistic weak simulations, and the other on a probabilistic extension of a fragment of Milner-Parrow-Walker modal logic for the $\pi$-calculus. We base our notions of simulations on similar concepts used in previous work for probabilistic CSP. However, unlike the case with CSP (or other non-value-passing calculi), there are several possible definitions of simulation for the probabilistic $\pi$-calculus, which arise from different ways of scoping the name quantification. We show that in order to capture the testing preorders, one needs to use the "earliest" simulation relation (in analogy to the notion of early (bi)simulation in the non-probabilistic case). The key ideas in both characterisations are the notion of a "characteristic formula" of a probabilistic process, and the notion of a "characteristic test" for a formula. As in an earlier work on testing equivalence for the $\pi$-calculus by Boreale and De Nicola, we extend the language of the $\pi$-calculus with a mismatch operator, without which the formulation of a characteristic test will not be possible.

**Keywords:** Probabilistic $\pi$-calculus; Testing semantics; Bisimulation; Modal logic

## 1. Introduction

We consider an extension of a finite version (without replication or recursion) of the $\pi$-calculus [17], a typical name-passing process calculus, with a probabilistic choice operator, alongside the non-deterministic choice operator of the $\pi$-calculus. Such an extension has been shown to be useful in modelling protocols and their properties, see, e.g., [19, 3]. The combination of both probabilistic and non-deterministic choice has long been a subject of study in process theories, see, e.g., [11, 28, 25, 24, 4, 20, 7, 1]. In this paper, we consider a natural notion of preorders for the probabilistic $\pi$-calculus, based on the notion of *testing* [5, 13]. In this

*Correspondence and offprint requests to*: yuxindeng@sjtu.edu.cn, Alwen.Tiu@anu.edu.au

testing theory, one defines a notion of test, what it means to apply a test to a process, the outcome of a test, and how the outcomes of tests can be compared. In general, the outcome of a test can be any non-empty set, endowed with a (partial) order; in the case of the original theory, this is simply a two-element lattice, with the top element representing success and the bottom element representing failure. In the probabilistic case, the set of outcomes is the unit interval [0,1], denoting probabilities of success, with the standard mathematical ordering $\leq$. In the presence of non-determinism, it is natural to consider a set of such probabilities as the result of applying a test to a process. Two standard approaches for comparing results of a test are the so-called Hoare preorder, written $\sqsubseteq_{Ho}$, and the Smyth preorder, $\sqsubseteq_{Sm}$ [12]:

- $O_1 \sqsubseteq_{Ho} O_2$ if for every $o_1 \in O_1$ there exists $o_2 \in O_2$ such that $o_1 \leq o_2$.
- $O_1 \sqsubseteq_{Sm} O_2$ if for every $o_2 \in O_2$ there exists $o_1 \in O_1$ such that $o_1 \leq o_2$.

Correspondingly, these give rise to two semantic preorders for processes:

- *may-testing*: $P \sqsubseteq_{pmay} Q$ iff for every test $T$, $\mathit{Apply}(T, P) \sqsubseteq_{Ho} \mathit{Apply}(T, Q)$
- *must-testing*: $P \sqsubseteq_{pmust} Q$ iff for every test $T$, $\mathit{Apply}(T, P) \sqsubseteq_{Sm} \mathit{Apply}(T, Q)$,

where $\mathit{Apply}(T, P)$ refers to the result of applying the test $T$ to process $P$.

We derive two characterisations of both may-testing and must-testing: one based on a notion of probabilistic weak (failure) simulation [25], and the other based on a modal logic obtained by extending Milner-Parrow-Walker (MPW) modal logic for the (non-probabilistic) $\pi$-calculus [18]. These characterisations are in part motivated by our desire to derive more operational notions of preorders that are amenable to automation. Both the testing preorders and the logic-based preorders involve quantification over test processess and logical formulas, respectively, and are thus unsuitable for mechanisation. The simulation preorders, on the other hand, avoid this quantification over tests or formulas, making them more suitable for mechanisation.

The probabilistic $\pi$-calculus that we consider here is a variant of the probabilistic $\pi$-calculus considered in [3], but extended with the mismatch operator. As has already been observed in the testing semantics for the non-probabilistic $\pi$-calculus [2], the omission of mismatch would result in a strictly less discriminating test. This is essentially due to the possibility of two kinds of output transitions in the $\pi$-calculus, a bound-output action, which outputs a new name, e.g., $\bar{x}(w).0$, and a free-output action, e.g., $\bar{x}y.0$. Without the mismatch operator, the two processes are related via may-testing, because the test cannot distinguish between output of a fresh name and output of an arbitrary name (see [2]).

The technical framework used to prove the main results in this paper is based on previous works on probabilistic CSP (pCSP) [9, 7], an extension of Hoare's CSP [15] with a probabilistic choice operator. This allows us to adapt some proofs and results from [9, 7] that are not calculus-specific. The name-passing feature of the $\pi$-calculus, however, gives rise to several difficulties not found in the non-name-passing calculi such as pCSP, and it consequently requires new techniques to deal with. For instance, there is not a canonical notion of (weak) simulation in the $\pi$-calculus, unlike the case with pCSP. Different variants arise from different ways of scoping the name quantification in the simulation clause dealing with input transitions, e.g., the "early" vs. the "late" variants of (bi)simulation [17]. In the case of weak simulation, one also gets a "delay" variant of (bi)simulation [10, 21, 27]. As we show in Section 4, the right notion of simulation is the early variant, as all other weak simulation relations are strictly more discriminating than the early one. Another difficulty is in proving congruence properties, a prerequisite for the soundness of the (failure) simulation preorders. The possibility of performing a 'close' communication in the $\pi$-calculus requires a combination of closure under parallel composition and name restriction (see Section 5). We use the so-called "up-to" techniques [22] for non-probabilistic calculi to prove these congruences.

We show that $\sqsubseteq_{pmay}$ coincides with a simulation preorder $\sqsubseteq_S$ and a preorder $\sqsubseteq_{\mathcal{L}}$ induced by a modal logic $\mathcal{L}$ extending the MPW logic. Dually, the must-testing preorder is shown to coincide with a failure simulation preorder, $\sqsubseteq_{FS}$, and a preorder $\sqsubseteq_{\mathcal{F}}$ induced by a modal logic $\mathcal{F}$ extending $\mathcal{L}$. For technical reasons in proving the completeness result of (failure) simulation, we make use of testing preorders involving vector-based testing ($\sqsubseteq^{\Omega}_{pmay}$ and $\sqsubseteq^{\Omega}_{pmust}$ below). The precise relations among these preorders are as follows (where we annotate the inclusions/equalities for ease of reference):

$$\begin{array}{lllllllll} \sqsubseteq_S & \subseteq^1 & \sqsubseteq_{pmay} & =^3 & \sqsubseteq^{\Omega}_{pmay} & \subseteq^5 & \sqsubseteq_{\mathcal{L}} & \subseteq^7 & \sqsubseteq_S \\ \sqsubseteq_{FS} & \subseteq^2 & \sqsubseteq_{pmust} & =^4 & \sqsubseteq^{\Omega}_{pmust} & \subseteq^6 & \sqsubseteq_{\mathcal{F}} & \subseteq^8 & \sqsubseteq_{FS} \end{array}$$

The (annotated) inclusions and equalities correspond to the following theorems: Theorem 3.3 (3 and 4), Theorem 5.23 (1 and 2), Theorem 6.7 (7 and 8), and Theorem 7.5 (5 and 6). The proofs of these inclusions are subjects of Section 5, Section 6 and Section 7. Let us highlight the characterisations of may-testing preorder. As with the case with pCSP [7], the key idea to the proof of the inclusion $\sqsubseteq_{\mathcal{L}} \subseteq \sqsubseteq_S$ is to show that for each process $P$, there exists a *characteristic formula* $\varphi_P$ such that if $Q \models \varphi_P$ then $P \sqsubseteq_S Q$. The inclusion $\sqsubseteq^{\Omega}_{pmay} \subseteq \sqsubseteq_{\mathcal{L}}$ is proved by showing that for each formula $\varphi$, there exists a *characteristic test* $T_\varphi$ such that for all process $P$, $P \models \varphi$ iff $P$ passes the test $T_\varphi$ with some threshold testing outcome.

## 2. Processes and probabilistic distributions

We consider an extension of the (finite) $\pi$-calculus with a probabilistic choice operator, ${}_p\oplus$, where $p \in (0, 1]$. We shall be using the late version of the operational semantics, formulated in the reactive style (in the sense of [26]) following previous work [9, 7]. The use of the late semantics allows for a straightforward definition of characteristic formulas (see Section 6), which are used in the completeness proof. So our testing equivalence is essentially a "late" testing equivalence. However, as has been shown in [16, 2], late and early testing equivalences coincide for value-passing/name-passing calculi.

We assume a countably infinite set of *names*, ranged over by $a, b, x, y$ etc. Given a name $a$, its *co-name* is $\bar{a}$. We use $\mu$ to denote a name or a co-name. Process expressions are generated by the following two-sorted grammar:

$$\begin{array}{rcl} P & ::= & s \mid P \,{}_p\!\oplus P \\ s & ::= & \mathbf{0} \mid a(x).P \mid \bar{a}x.P \mid [x = y]s \mid [x \neq y]s \mid s + s \mid s|s \mid \nu x.s \end{array}$$

We let $P, Q, ...$ range over process terms defined by this grammar, and $s, t$ range over the subset $S_p$ comprising only the state-based process terms, i.e. the sub-sort $s$.

The input prefix $a(x)$ and restriction $\nu x$ are name-binding contructs; $x$ in this case is a bound name. We denote with $fn(P)$ the set of free names in $P$ and $bn(P)$ the set of bound names. The set of names in $P$ (free or bound) is denoted by $n(P)$. We shall assume that bound names are different from each other and different from any free names. Processes are considered equivalent modulo renaming of bound names. Processes are ranged over by $P$,$Q$,$R$, etc. We shall refer to our probabilistic extension of the $\pi$-calculus as $\pi_p$.

We shall sometimes use an $n$-ary version of the binary operators. For example, we use $\bigoplus_{i\in I} p_i P_i$, where $\sum_{i\in I} p_i = 1$, to denote a process obtained by several applications of the probabilistic choice operator. Similarly, $\sum_{i\in I} P_i$ denotes several applications of the non-deterministic choice operator $+$. We shall use the $\tau$-prefix, as in $\tau.P$, as an abbreviation of $\nu x(x(y).\mathbf{0} \mid \bar{x}x.P)$, where $x, y \notin fn(P)$.

In this paper, we take the viewpoint that a probabilistic process represents an unstable state that may probabilistically evolve into some stable states. Formally, we describe unstable states as distributions and stable states as state-based processes. Note that in a state-based process, probabilistic choice can only appear under input/output prefixes. The operational semantics of $\pi_p$ will be defined only for state-based processes.

Probabilistic distributions are ranged over by $\Delta$. A *discrete probabilistic distribution* over a set $S$ is a mapping $\Delta : S \to [0, 1]$ with $\sum_{s\in S} \Delta(s) = 1$. The *support* of a distribution $\Delta$, denoted by $\lceil\Delta\rceil$, is the set $\{s \mid \Delta(s) > 0\}$. From now on, we shall restrict to only probabilistic distributions with finite support, and we let $\mathcal{D}(S)$ denote the collection of such distributions over $S$. If $s$ is a state-based process, then $\delta[s]$ denote the point distribution that maps $s$ to 1. For a finite index set $I$, given $p_i$ and distribution $\Delta_i$, for each $i \in I$, such that $\sum_{i\in I} p_i = 1$, we define another probability distribution $\sum_{i\in I} p_i \cdot \Delta_i$ as $(\sum_{i\in I} p_i \cdot \Delta_i)(s) = \sum_{i\in I} p_i \cdot \Delta_i(s)$, where $\cdot$ here denotes multiplication. We shall sometimes write this distribution as a summation $p_1 \cdot \Delta_1 + p_2 \cdot \Delta_2 + \ldots + p_n \cdot \Delta_n$ when the index set $I$ is $\{1, \ldots, n\}$.

A *probabilistic labelled transition system* (pLTS) is a triple $\langle S, L, \rightarrow\rangle$, where $S$ is a set of states, $L$ is a set of labels, and the transition relation $\rightarrow$ is a subset of $S \times L \times \mathcal{D}(S)$. We usually write $s \xrightarrow{\alpha} \Delta$ for $(s, \alpha, \Delta) \in \rightarrow$.

Probabilistic processes are interpreted as distributions over state-based processes as follows.

$$\begin{array}{rcl} [\![s]\!] & ::= & \delta[s] \text{ for } s \in S_p \\ [\![P \,{}_p\!\oplus Q]\!] & ::= & p \cdot [\![P]\!] + (1 - p) \cdot [\![Q]\!] \end{array}$$

Note that for each process term $P$ the distribution $[\![P]\!]$ is finite, that is it has finite support.

$$\frac{}{\alpha.P \xrightarrow{\alpha} [\![P]\!]}\ \textbf{Act} \qquad \frac{s \xrightarrow{\alpha} \Delta}{s + t \xrightarrow{\alpha} \Delta}\ \textbf{Sum}$$

$$\frac{s \xrightarrow{\alpha} \Delta}{[x = x]s \xrightarrow{\alpha} \Delta}\ \textbf{Match} \qquad \frac{s \xrightarrow{\alpha} \Delta}{[x \neq y]s \xrightarrow{\alpha} \Delta}\ \textbf{Mismatch}, x \neq y$$

$$\frac{s \xrightarrow{\alpha} \Delta}{s \mid t \xrightarrow{\alpha} \Delta \mid \delta[t]}\ \textbf{Par}, bn(\alpha) \cap fn(t) = \emptyset$$

$$\frac{s \xrightarrow{a(x)} \Delta_1 \quad t \xrightarrow{\bar{a}y} \Delta_2}{s \mid t \xrightarrow{\tau} \Delta_1[y/x] \mid \Delta_2}\ \textbf{Com} \qquad \frac{s \xrightarrow{a(w)} \Delta_1 \quad t \xrightarrow{\bar{a}(w)} \Delta_1}{s \mid t \xrightarrow{\tau} \nu w.(\Delta_1 \mid \Delta_2)}\ \textbf{Close}$$

$$\frac{s \xrightarrow{\alpha} \Delta}{\nu x.s \xrightarrow{\alpha} \nu x.\Delta}\ \textbf{Res}, x \notin n(\alpha) \qquad \frac{s \xrightarrow{\bar{x}z} \Delta}{\nu z.s \xrightarrow{\bar{x}(y)} \Delta[y/z]}\ \textbf{Open}, y \neq x, y \notin fn(\nu z.s)$$

**Fig. 1.** The late operational semantics of $\pi_p$.

A transition judgment can take one of the following forms:

$$s \xrightarrow{a(x)} \Delta \qquad s \xrightarrow{\tau} \Delta \qquad s \xrightarrow{\bar{a}x} \Delta \qquad s \xrightarrow{\bar{a}(x)} \Delta$$

The action $a(x)$ is called a *bound-input action*; $\tau$ is the silent action; $\bar{a}x$ is a *free-output action* and $\bar{a}(x)$ is a *bound-output action*. In actions $a(x)$ and $\bar{a}(x)$, $x$ is a bound name. Actions are ranged over by $\alpha$. Given an action $\alpha$, we denote with $fn(\alpha)$ the set of free names in $\alpha$, i.e., those names in $\alpha$ which are not bound names. The set of bound names in $\alpha$ is denoted by $bn(\alpha)$, and the set of all names (free and bound) in $\alpha$ is denoted by $n(\alpha)$. The free names of a distribution is the union of free names of its support, i.e., $fn(\Delta) = \bigcup\{fn(s) \mid s \in \lceil\Delta\rceil\}$.

A substitution is a mapping from names to names; substitutions are ranged over by $\rho, \sigma$ and $\theta$. A substitution $\theta$ is a *renaming substitution* if $\theta$ is an injective map, i.e., $\theta(x) = \theta(y)$ implies $x = y$. A substitution is extended to a mapping between processes in the standard way, avoiding capture of free variables. We use the notation $s[y/x]$ to denote the result of substituting free occurrences of $x$ in $s$ with $y$. Substitution is lifted to a mapping between distributions as follows:

$$\Delta[y/x](s) = \sum\{\Delta(s') \mid s'[y/x] = s\}.$$

It can be verified that $[\![P[y/x]]\!] = [\![P]\!][y/x]$ for every process $P$.

The operational semantics of state-based processes is given in terms of a pLTS where the set of states is $S_p$ and the transition relation is generated by the rules in Figure 1. The rules for parallel composition and restriction use an obvious notation for distributing an operator over distributions, for example:

$$(\Delta_1 \mid \Delta_2)(s) = \begin{cases} \Delta_1(s_1) \cdot \Delta_2(s_2) & \text{if } s = s_1|s_2 \\ 0 & \text{otherwise} \end{cases}$$

$$(\nu x.\Delta)(s) = \begin{cases} \Delta(s') & \text{if } s = \nu x.s' \\ 0 & \text{otherwise.} \end{cases}$$

The symmetric counterparts of **Sum**, **Par**, **Com** and **Close** are omitted.

## 3. Testing probabilistic processes

As standard in testing theories [5, 13, 2], to define a test, we introduce a distinguished name $\omega$ which can only be used in tests and is not part of the processes being tested. A *test* is just a probabilistic process with possible free occurrences of the name $\omega$ as channel name in output prefixes, i.e., a test is a process which

may have subterms of the form $\bar{\omega}a.P$. Note that the object of the action prefix (i.e., the name $a$) is irrelevant for the purpose of testing. For simplicity, we shall assume that $a$ is $\omega$, i.e., a successful test will output on channel $\omega$ the same channel name. Note also that it makes no differences whether the name $\omega$ appears in input prefixes instead of output prefixes; the notion of testing preorder will remain the same. Therefore we shall often simply write $\omega.P$ to denote $\bar{\omega}\omega.P$, and $P \xrightarrow{\omega} \Delta$ to denote $P \xrightarrow{\bar{\omega}\omega} \Delta$. The definitions of may-testing preorder, $\sqsubseteq_{pmay}$, and must-testing preorder, $\sqsubseteq_{pmust}$, have already been given in the introduction, but we left out the definition of the *Apply* function. This will be given below.

Following [9], to define the *Apply* function, we first define a *results-gathering function* $\mathbb{V} : S_p \to \mathcal{P}([0,1])$:

$$\mathbb{V}(s) = \begin{cases} \{1\} & \text{if } s \xrightarrow{\omega} \\ \bigcup\{\mathbb{V}(\Delta) \mid s \xrightarrow{\tau} \Delta\} & \text{if } s \not\xrightarrow{\omega} \text{ but } s \xrightarrow{\tau} \\ \{0\} & \text{otherwise.} \end{cases}$$

Here the notation $\mathcal{P}([0,1])$ stands for the powerset of $[0,1]$, and we use $\mathbb{V}(\Delta)$ to denote the set of probabilities $\{\sum_{s\in\lceil\Delta\rceil} \Delta(s)\cdot p_s \mid p_s \in \mathbb{V}(s)\}$. The *Apply* function is defined as follows: given a test $T$ and a process $P$,

$$Apply(T,P) = \mathbb{V}([\![\nu\vec{x}.(T \mid P)]\!])$$

where $\{\vec{x}\}$ is the set of free names in $T$ and $P$, excluding $\omega$. So the process (or rather, the distribution) $\nu\vec{x}.(T \mid P)$ can only perform an observable action on $\omega$.

As the definition of testing preorders involves quantification over tests, in general it is difficult to establish directly that two processes are related by these preorders. However, showing that they are *not* related by the preorders is easier, i.e., one needs only to demonstrate a test that distinguishes them.

**Example 3.1.** Let $P = a(x).\bar{a}c$ and $Q = a(x).[x = b]\bar{a}c \,{}_{\frac{1}{2}}\!\oplus a(x).[x \neq b]\bar{a}c$. We claim that $P \not\sqsubseteq_{pmust} Q$ and $P \not\sqsubseteq_{pmay} Q$. To see why, consider the test $T = \bar{a}b.a(x).\omega$. Then we have $Apply(T,P) = \{1\}$ but $Apply(T,Q) = \{\frac{1}{2}\}$.

**Vector-based testing.** Following [7], we introduce another approach of testing called *vector-based testing*, which will play an important role in Section 7.

Let $\Omega$ be a set of fresh success actions different from any normal channel names. An $\Omega$-test is a $\pi_p$-process, but allowing subterms $\omega.P$ for any $\omega \in \Omega$. Applying such a test $T$ to a process $P$ yields a non-empty set of test outcome-tuples $Apply^{\Omega}(T,P) \subseteq [0,1]^{\Omega}$. For each such tuple, its $\omega$-component gives the probability of successfully performing action $\omega$.

To define a results-gathering function for vector-based testing, we need some auxiliary notations. For any action $\alpha$ define $\alpha! : [0,1]^{\Omega} \to [0,1]^{\Omega}$ by

$$\alpha!o(\omega) = \begin{cases} 1 & \text{if } \omega = \alpha \\ o(\omega) & \text{otherwise} \end{cases}$$

so that if $\alpha$ is a success action in $\Omega$ then $\alpha!$ updates the tuple 1 at that point, leaving it unchanged otherwise, and when $\alpha \notin \Omega$ the function $\alpha!$ is the identity. For any set $O \subseteq [0,1]^{\Omega}$, we write $\alpha!O$ for the set $\{\alpha!o \mid o \in O\}$. For any set $X$ define its *convex closure* $\updownarrow X$ by

$$\updownarrow X \;:=\; \{\sum_{i\in I} p_i \cdot o_i \mid o_i \in X \text{ for each } i \in I \text{ and } \textstyle\sum_{i\in I} p_i = 1\}.$$

Here, $I$ is assumed to be a finite index set. Finally, zero vector $\vec{0}$ is given by $\vec{0}(\omega) = 0$ for all $\omega \in \Omega$. Let $S_p^{\Omega}$ be the set of state-based $\Omega$-tests.

**Definition 3.2.** The vector-based results-gathering function $\mathbb{V}^{\Omega} : S_p^{\Omega} \to \mathcal{P}([0,1]^{\Omega})$ is given by

$$\mathbb{V}^{\Omega}(s) \;:=\; \begin{cases} \updownarrow\bigcup\{\alpha!(\mathbb{V}^{\Omega}(\Delta)) \mid s \xrightarrow{\alpha} \Delta\} & \text{if } s \to \\ \{\vec{0}\} & \text{otherwise} \end{cases}$$

The notation $s \to$ means that $s$ is not a deadlock state, i.e. there is some $\alpha$ and $\Delta$ such that $s \xrightarrow{\alpha} \Delta$. For any process $P$ and $\Omega$-test $T$, we define $Apply^{\Omega}(T,P)$ as $\mathbb{V}^{\Omega}([\![\nu\vec{x}.(T|P)]\!])$, where $\{\vec{x}\} = fn(T,P) - \Omega$. The

vector-based may and must preorders are given by

$$P \sqsubseteq^{\Omega}_{pmay} Q \quad \text{iff} \quad \text{for all } \Omega\text{-test } T : \mathit{Apply}^{\Omega}(T,P) \sqsubseteq_{Ho} \mathit{Apply}^{\Omega}(T,Q)$$
$$P \sqsubseteq^{\Omega}_{pmust} Q \quad \text{iff} \quad \text{for all } \Omega\text{-test } T : \mathit{Apply}^{\Omega}(T,P) \sqsubseteq_{Sm} \mathit{Apply}^{\Omega}(T,Q)$$

where $\sqsubseteq_{Ho}$ and $\sqsubseteq_{Sm}$ are the Hoare and Smyth preorders on $\mathcal{P}([0,1]^{\Omega})$ generated from $\leq$ index-wise on $[0,1]^{\Omega}$.

Notice a subtle difference between the definition of $\mathbb{V}^{\Omega}$ above and the definition of $\mathbb{V}$ given earlier. In $\mathbb{V}^{\Omega}$, we use *action-based testing*, i.e., the actual execution of $\omega$ constitutes a success. This is in contrast to the *state-based testing* in $\mathbb{V}$, where a success is defined for a state where a success action $\omega$ is possible, without having to actually perform the action $\omega$. In the case where there is no divergence, as in our case, these two notions of testing coincide; see [7] for more details.

The following theorem can be shown by adapting the proof of Theorem 6.6 in [7], which states a general property about pLTSs [6].

**Theorem 3.3.** Let $P$ and $Q$ be any $\pi_p$-processes.

1. $P \sqsubseteq^{\Omega}_{pmay} Q$ iff $P \sqsubseteq_{pmay} Q$
2. $P \sqsubseteq^{\Omega}_{pmust} Q$ iff $P \sqsubseteq_{pmust} Q$.

## 4. Simulation and Failure Simulation

To define (failure) simulation, we need to generalise the transition relations between states and distributions to those between distributions and distributions. This is defined via a notion of lifting of a relation.

**Definition 4.1 (Lifting [8]).** Given a relation $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$, define a *lifted relation* $\overline{\mathcal{R}} \subseteq \mathcal{D}(S_p) \times \mathcal{D}(S_p)$ as the smallest relation that satisfies

1. $s\mathcal{R}\Theta$ implies $\delta[s]\ \overline{\mathcal{R}}\ \Theta$
2. (Linearity) $\Delta_i\ \overline{\mathcal{R}}\ \Theta_i$ for all $i \in I$ implies $(\sum_{i\in I} p_i{\cdot}\Delta_i)\ \overline{\mathcal{R}}\ (\sum_{i\in I} p_i{\cdot}\Theta_i)$ for any $p_i \in [0,1]$ with $\sum_{i\in I} p_i = 1$.

The following is a useful properties of the lifting operation.

**Proposition 4.2 ([9]).** Suppose $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ and $\sum_{i\in I} p_i = 1$. If $(\sum_{i\in I} p_i{\cdot}\Delta_i)\ \overline{\mathcal{R}}\ \Theta$ then $\Theta = \sum_{i\in I} p_i{\cdot}\Theta_i$ for some set of distributions $\Theta_i$ such that $\Delta_i\ \overline{\mathcal{R}}\ \Theta_i$ for all $i \in I$.

For simplicity of presentation, the lifted version of the transition relation $\xrightarrow{\alpha}$ will be denoted by the same notation as the unlifted version. So we shall write $\Delta \xrightarrow{\alpha} \Theta$ when $\Delta$ and $\Theta$ are related by the lifted relation from $\xrightarrow{\alpha}$ . Note that in the lifted transition $\Delta \xrightarrow{\alpha} \Theta$, *all* processes in $\lceil\Delta\rceil$ must be able to simultaneously make the transition $\alpha$. For example,

$$\frac{1}{2}\cdot\delta[\bar{a}x.s] + \frac{1}{2}\cdot\delta[\bar{a}x.t] \xrightarrow{\bar{a}x} \frac{1}{2}\cdot\delta[s] + \frac{1}{2}\cdot\delta[t]$$

but the distribution $\frac{1}{2}\cdot\delta[\bar{a}x.s] + \frac{1}{2}\cdot\delta[\bar{b}x.t]$ will not be able to make that transition. We need a few more relations to define (failure) simulation:

- We write $s \xrightarrow{\hat{\tau}} \Delta$ to denote either $s \xrightarrow{\tau} \Delta$ or $\Delta = \delta[s]$. Its lifted version will be denoted by the same notation, e.g., $\Delta_1 \xrightarrow{\hat{\tau}} \Delta_2$. The reflexive-transitive closure of the latter is denoted by $\stackrel{\hat{\tau}}{\Longrightarrow}$ .
- $\Delta_1 \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta_2$, for $\alpha \neq \tau$, iff $\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta' \xrightarrow{\alpha} \Delta'' \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_2$ for some $\Delta'$ and $\Delta''$.
- We write $s \downarrow_a$ to denote $s \xrightarrow{a(x)}$, and $s \downarrow_{\bar{a}}$ to denote either $s \xrightarrow{\bar{a}(x)}$ or $s \xrightarrow{\bar{a}x}$; $s \not\downarrow_{\mu}$ stands for the negation. We write $s \not\downarrow_X$ when $s \not\xrightarrow{\tau}$ and $\forall \mu \in X : s \not\downarrow_{\mu}$, and $\Delta \not\downarrow_X$ when $\forall s \in \lceil\Delta\rceil : s \not\downarrow_X$.

**Definition 4.3.** A relation $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$ is said to be a *failure simulation* if $s\mathcal{R}\Theta$ implies:

1. If $s \xrightarrow{a(x)} \Delta$ and $x \notin fn(s,\Theta)$, then for every name $w$, there exists $\Theta_1$, $\Theta_2$ and $\Theta'$ such that
$$\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta_1 \xrightarrow{a(x)} \Theta_2, \qquad \Theta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta', \qquad \text{and} \qquad (\Delta[w/x]) \ \overline{\mathcal{R}} \ \Theta'.$$
2. If $s \xrightarrow{\alpha} \Delta$, where $\alpha$ is not an input action, and $bn(\alpha) \notin fn(s,\Theta)$, then there exists $\Theta'$ such that $\Theta \stackrel{\hat{\alpha}}{\Longrightarrow} \Theta'$ and $\Delta \ \overline{\mathcal{R}} \ \Theta'$
3. If $s \not\downarrow_X$ then there exists $\Theta'$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \not\downarrow_X$.

We denote with $\lhd_{FS}$ the largest failure simulation relation. Similarly, we define *simulation* and $\lhd_S$ by dropping the third clause above. The *simulation preorder* $\sqsubseteq_S$ and *failure simulation preorder* $\sqsubseteq_{FS}$ on process terms are defined by letting

$$P \sqsubseteq_S Q \text{ iff there is a distribution } \Theta \text{ with } [\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \text{ and } [\![P]\!] \ \overline{\lhd_S} \ \Theta.$$
$$P \sqsubseteq_{FS} Q \text{ iff there is a distribution } \Theta \text{ with } [\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \text{ and } [\![Q]\!] \ \overline{\lhd_{FS}} \ \Theta.$$

Below is a simple example that illustrates the use of the simulation preorder.

**Example 4.4.** Let $P = a(x).(\bar{a}c \,{}_{\frac{1}{2}}\!\oplus 0)$ and $Q = a(x).[x = b]\bar{a}c \,{}_{\frac{1}{2}}\!\oplus a(x).[x \neq b]\bar{a}c$. We claim that $P \sqsubseteq_S Q$. To prove this, it is enough to show that there is a simulation relation $\mathcal{R}$ such that $[\![P]\!] \ \overline{\mathcal{R}} \ [\![Q]\!]$. Let $\mathcal{R}$ be the smallest set containing the following pairs:

$$P \ \mathcal{R} \ [\![Q]\!], \qquad \bar{a}c \ \mathcal{R} \ \delta[[b = b]\bar{a}c], \qquad 0 \ \mathcal{R} \ \delta[[b \neq b]\bar{a}c], \qquad 0 \ \mathcal{R} \ 0,$$

and, for every $y$ distinct from $b$: $\bar{a}c \ \mathcal{R} \ \delta[[y \neq b]\bar{a}c]$ and $0 \ \mathcal{R} \ \delta[[y = b]\bar{a}c]$. Because $P \ \mathcal{R} \ [\![Q]\!]$ (note that $P$ is a state-based process), it is immediate that $[\![P]\!] \ \overline{\mathcal{R}} \ [\![Q]\!]$ by Definition 4.1. It remains to show that $\mathcal{R}$ is indeed a simulation. This is easily checked by following through Definition 4.3. For example, since $P \ \mathcal{R} \ [\![Q]\!]$ and the input action is possible from $P$, we have to show its continuations satisfies clause (1) in Definition 4.3. We have that $P \xrightarrow{a(x)} (\frac{1}{2} \cdot \bar{a}c + \frac{1}{2} \cdot 0)$, and also:

$$[\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} [\![Q]\!] \xrightarrow{a(x)} (\frac{1}{2} \cdot [x = b]\bar{a}c + \frac{1}{2} \cdot [x \neq b]\bar{a}c) = \Theta.$$

It remains to show that for every $w$, $(\frac{1}{2} \cdot \bar{a}c + \frac{1}{2} \cdot 0) \ \overline{\mathcal{R}} \ \Theta[w/x]$. If $w = b$, then this follows from the fact that $\bar{a}c \ \mathcal{R} \ \delta[[b = b]\bar{a}c]$ and $0 \ \mathcal{R} \ \delta[[b \neq b]\bar{a}c]$. Otherwise, $w \neq b$ and it follows from $\bar{a}c \ \mathcal{R} \ \delta[[w \neq b]\bar{a}c]$ and $0 \ \mathcal{R} \ \delta[[w = b]\bar{a}c]$.

Notice the rather unusual clause for input action in Definition 4.3, where no silent action from $\Theta_2$ is permitted after the input transition. This is reminiscent of the notion of *delay (bi)simulation* [10, 21, 27]. If instead of that clause, we simply require $\Theta \stackrel{\widehat{a(x)}}{\Longrightarrow} \Theta''$ and $\Delta[w/x] \ \overline{\mathcal{R}} \ \Theta''[w/x]$ then, in the presence of mismatch, simulation is not sound w.r.t. the may-testing preorder, even in the non-probabilistic case. Consider, for example, the following processes:

$$P = a(x).\bar{a}b \qquad Q = a(x).[x \neq c]\tau.\bar{a}b$$

where we recall that $\tau.R$ abbreviates $\nu z.(z(u) \mid \bar{z}z.R)$ for some $z \notin fn(R)$. The process $P$ can make an input transition, and regardless of the value of $x$, it can then output $b$ on channel $a$. Notice that for $Q$, we have

$$Q \xrightarrow{a(x)} [x \neq c]\tau.\bar{a}b \xrightarrow{\tau} \nu z(0 \mid \bar{a}b) = Q'.$$

$Q'$ can also output $b$ on channel $a$, so under this alternative definition, $Q$ can simulate $P$. But $P \not\sqsubseteq_{pmay} Q$, as the test $\bar{a}c.a(y).\omega$ will distinguish them. This issue has also appeared in the theory of weak (late) bisimulation for the non-probabilistic $\pi$-calculus; see, e.g., [23].

Note that the above definition of $\lhd_S$ is what is usually called the "early" simulation. One can obtain different variants of "late" simulation using different alternations of the universal quantification on names and the existential quantifications on distributions in clause 1 of Definition 4.3. Any of these variants leads to a strictly more discriminating simulation. To see why, consider the weaker of such late variants, i.e., one in which the universal quantifier on $w$ comes after the existential quantifier on $\Theta_1$:

*If* $s \xrightarrow{a(x)} \Delta$ *and* $x \notin fn(s, \Theta)$*, then there exists* $\Theta_1$ *such that for every name* $w$*, there exist* $\Theta_2$ *and* $\Theta'$ *such that* $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta_1 \xrightarrow{a(x)} \Theta_2$*,* $\Theta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$*, and* $(\Delta[w/x])\ \overline{\mathcal{R}}\ \Theta'$*.*
Let us denote this variant with $\sqsubseteq_{S'}$ . Consider the following processes:

$$P = a(x).\bar{b}x.\mathbf{0} + a(x).\mathbf{0} + a(x).[x = z]\bar{b}x.\mathbf{0} \qquad Q = \tau.a(x).\bar{b}x.\mathbf{0} + \tau.a(x).\mathbf{0}$$

It is easy to see that $P \sqsubseteq_S Q$ but $P \not\sqsubseteq_{S'} Q$.

If we drop the silent transitions $\Theta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ in clause (1) of Definition 4.3, i.e., we let $\Theta' = \Theta_2[w/x]$ (hence, we get a delay simulation), then again we get a strictly stronger relation than $\sqsubseteq_S$. Let us refer to this stronger relation as $\sqsubseteq_D$. Let $P$ be $a(x).(c\,_{\frac{1}{2}}\!\oplus d)$ and let $Q$ be $a(x).\tau.(c\,_{\frac{1}{2}}\!\oplus d)$. Here we remove the parameters in the input prefixes $c$ and $d$ to simplify presentation. Again, it can be shown that $P \sqsubseteq_S Q$ but $P \not\sqsubseteq_D Q$. For the latter to hold, we would have to prove $\frac{1}{2} \cdot \delta[c] + \frac{1}{2} \cdot \delta[d]\ \overline{\lhd_S}\ \delta[\tau.(c\,_{\frac{1}{2}}\!\oplus d)]$, which is impossible.

Note that (failure) simulation is a relation between processes and distributions, rather than between processes, so it is not immediately obvious that it is a preorder. This is established in Corollary 4.14 below, whose proof requires a series of lemmas.

In the following, when we apply a substitution to an action, we assume that the substitution affects both the free and the bound names in the action. For example, if $\alpha = a(x)$ and $\theta = [b/a, y/x]$ then $\alpha\theta = b(y)$. However, application of a substitution to processes or distributions must still avoid capture.

**Lemma 4.5.** Suppose $\sigma$ is a renaming substitution.

1. If $s \xrightarrow{\alpha} \Delta$ then $s\sigma \xrightarrow{\alpha\sigma} \Delta\sigma$.
2. If $\Delta \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta'$ then $\Delta\sigma \stackrel{\hat{\alpha}\sigma}{\Longrightarrow} \Delta'\sigma$.

**Lemma 4.6.** Let $I$ be a finite index set, and let $\sum_{i\in I} p_i = 1$. Suppose $s_i \xrightarrow{a(x_i)} \Delta_i$ for each $i \in I$. Let $x$ be a fresh name not occuring in any of $s_i$, $a(x_i)$ or $\Delta_i$. Then

$$\sum_{i\in I} p_i \cdot \delta[s_i] \xrightarrow{a(x)} \sum_{i\in I} p_i \cdot \Delta_i[x/x_i].$$

Given the above lemma, given transitions $s_i \xrightarrow{a(x_i)} \Delta_i$, we can always assume that, all the $x_i$'s are the same fresh name, so that when lifting those transitions to distributions, we shall omit the explicit renaming of individual $x_i$. This will simplify the presentation of the proofs in the following. The same remark applies to bound output transitions.

**Lemma 4.7.** Suppose $\sum_{i\in I} p_i = 1$ and $\Delta_i \stackrel{\hat{\alpha}}{\Longrightarrow} \Phi_i$ for each $i \in I$, where $I$ is a finite index set. Then

$$\sum_{i\in I} p_i \cdot \Delta_i \stackrel{\hat{\alpha}}{\Longrightarrow} \sum_{i\in I} p_i \cdot \Phi_i.$$

*Proof.* Same as in the proof of Lemma 6.6. in [9]. □

**Lemma 4.8.** For every state-based process $s$, we have $s \lhd_S \delta[s]$ and $s \lhd_{FS} \delta[s]$.

*Proof.* Let $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$ be the relation defined as follows: $s\ \mathcal{R}\ \Theta$ iff $\Theta = \delta[s]$. It is easy to see that $\mathcal{R}$ is a simulation and also a failure simulation. □

**Lemma 4.9.** Suppose $\Delta\ \overline{\lhd}_S\ \Phi$ and $\Delta \xrightarrow{\alpha} \Delta'$, where $\alpha$ is either $\tau$, a free action or a bound output action. Then $\Phi \xrightarrow{\hat{\alpha}} \Phi'$ for some $\Phi'$ such that $\Delta'\ \overline{\lhd}_S\ \Phi'$.

*Proof.* Similar to the proof of Lemma 6.7 in [9]. □

**Lemma 4.10.** Suppose $\Delta\ \overline{\lhd}_S\ \Phi$ and $\Delta \xrightarrow{a(x)} \Delta'$. Then for all name $w$, there exist $\Psi_1$, $\Psi_2$ and $\Psi$ such that

$$\Phi \stackrel{\hat{\tau}}{\Longrightarrow} \Psi_1 \xrightarrow{a(x)} \Psi_2, \qquad \Psi_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Psi, \qquad \text{and} \qquad (\Delta'[w/x])\ \overline{\lhd}_S\ \Psi.$$

*Proof.* From $\Delta \mathrel{\overline{\triangleleft}_S} \Phi$ we have that

$$\Delta = \sum_{i\in I} p_i \cdot \delta[s_i], \qquad s_i \mathrel{\triangleleft_S} \Phi_i, \qquad \Phi = \sum_{i\in I} p_i \cdot \Phi_i. \tag{1}$$

and from $\Delta \xrightarrow{a(x)} \Delta'$ we have:

$$\Delta = \sum_{j\in J} q_j \cdot \delta[t_j], \qquad t_j \xrightarrow{a(x)} \Theta_j, \qquad \Delta' = \sum_{j\in J} q_j \cdot \Theta_j. \tag{2}$$

We assume w.l.o.g. that all $p_i$ and $q_j$ are non-zero. Following [9], we define two index sets: $I_j = \{i \in I \mid s_i = t_j\}$ and $J_i = \{j \in J \mid t_j = s_i\}$. Obviously, we have

$$\{(i,j) \mid i \in I, j \in J_i\} = \{(i,j) \mid j \in J, i \in J_i\}, \quad \text{and} \tag{3}$$

$$\Delta(s_i) = \sum_{j\in J_i} q_j \qquad \Delta(t_j) = \sum_{i\in I_j} p_i. \tag{4}$$

It follows from (4) that we can rewrite $\Phi$ as

$$\Phi = \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Phi_i.$$

Note that $s_i = t_j$ when $j \in I_i$. Since $s_i \mathrel{\triangleleft_S} \Phi_i$, and $s_i = t_j \xrightarrow{a(x)} \Theta_j$, we have, given any name $w$, some $\Phi^1_{ij}$, $\Phi^2_{ij}$ and $\Phi_{ij}$ such that:

$$\Phi_i \stackrel{\hat{\tau}}{\Longrightarrow} \Phi^1_{ij} \xrightarrow{a(x)} \Phi^2_{ij}, \qquad \Phi^2_{ij}[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_{ij}, \qquad \Theta_j[w/x] \mathrel{\overline{\triangleleft}_S} \Phi_{ij}. \tag{5}$$

Let

$$\Psi_1 = \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Phi^1_{ij} \qquad \Psi_2 = \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Phi^2_{ij} \qquad \Psi = \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Phi_{ij}.$$

Lemma 4.7 and (5) above give us:

$$\Phi = \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Phi_i \stackrel{\hat{\tau}}{\Longrightarrow} \Psi_1 \xrightarrow{a(x)} \Psi_2 \qquad \Psi_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Psi$$

It remains to show that $\Delta'[w/x] \mathrel{\overline{\triangleleft}_S} \Psi$.

$$\begin{aligned}
\Delta'[w/x] &= \sum_{j\in J} q_j \cdot \Theta_j[w/x] \\
&= \sum_{j\in J} q_j \cdot \sum_{i\in I_j} \frac{p_i}{\Delta(t_j)} \cdot \Theta_j[w/x] && \text{using (4)} \\
&= \sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(s_i)} \cdot \Theta_j[w/x] && \text{using (3)} \\
\mathrel{\overline{\triangleleft}_S} &\sum_{i\in I} \sum_{j\in J_i} \frac{p_i \cdot q_j}{\Delta(t_j)} \cdot \Phi_{ij} = \Psi && \text{using (5) and linearity of } \mathrel{\overline{\triangleleft}_S}
\end{aligned}$$

□

**Lemma 4.11.** Suppose $\Delta \mathrel{\overline{\triangleleft}_S} \Phi$ and $\Delta \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta'$, where $\alpha$ is either $\tau$, a free action or a bound output. Then $\Phi \stackrel{\hat{\alpha}}{\Longrightarrow} \Phi'$ for some $\Phi'$ such that $\Delta' \mathrel{\overline{\triangleleft}_S} \Phi'$.

*Proof.* Similar to the proof of Lemma 6.8 in [9]. □

**Proposition 4.12.** The relation $\mathrel{\overline{\triangleleft}_S}$ is reflexive and transitive.

*Proof.* Reflexivity of $\overline{\lhd}_S$ follows from Lemma 4.8. To show transitivity, let us define a relation $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$ as follows: $s \ \mathcal{R} \ \Theta$ iff there exists $\Delta$ such that $s \ \lhd_S \ \Delta$ and $\Delta \ \overline{\lhd}_S \ \Theta$. We show that $\mathcal{R}$ is a simulation.

But first, we claim that $\Theta \ \overline{\lhd}_S \ \Delta \ \overline{\lhd}_S \ \Phi$ implies $\Theta \ \overline{\mathcal{R}} \ \Phi$. This can be proved similarly as in the case of CSP (see the proof of Proposition 6.9 in [9]).

Now to show that $\mathcal{R}$ is a simulation, there are two cases to consider. Suppose $s \ \mathcal{R} \ \Phi$, i.e., $s \ \lhd_S \ \Delta \ \overline{\lhd}_S \ \Phi$.

- Suppose $s \xrightarrow{\alpha} \Theta$, where $\alpha$ is either $\tau$, a free action or a bound output action. From $s \ \lhd_S \ \Delta$, we have

$$\Delta \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta' \qquad \text{and} \qquad \Theta \ \overline{\lhd}_S \ \Delta'. \tag{6}$$

  By Lemma 4.11 and (6), we have $\Phi \stackrel{\hat{\alpha}}{\Longrightarrow} \Phi'$ and $\Delta' \ \overline{\lhd}_S \ \Phi'$, and by the above claim and (6), $\Theta \ \overline{\mathcal{R}} \ \Phi'$.

- Suppose $s \xrightarrow{a(x)} \Theta$, so we have: for all $w$, there exist $\Delta_1$, $\Delta_2$, and $\Delta'$ such that

$$\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \xrightarrow{a(x)} \Delta_2, \qquad \Delta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta', \qquad \text{and } \Theta[w/x] \ \overline{\lhd}_S \ \Delta'. \tag{7}$$

  Since $\Delta \ \overline{\lhd}_S \ \Phi$, by Lemma 4.11 we have $\Phi \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_1$ and $\Delta_1 \ \overline{\lhd}_S \ \Phi_1$. And since $\Delta_1 \xrightarrow{a(x)} \Delta_2$, by Lemma 4.10, for all $w$, there exist $\Phi_2$, $\Phi_3$ and $\Phi_4$ such that:

$$\Phi_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_2 \xrightarrow{a(x)} \Phi_3, \qquad \Phi_3[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_4, \qquad \Delta_2[w/x] \ \overline{\lhd}_S \ \Phi_4.$$

  Lemma 4.11, together with $\Delta_2[w/x] \ \overline{\lhd}_S \ \Phi_4$ and $\Delta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$, implies that $\Phi_4 \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_5$ and $\Delta' \ \overline{\lhd}_S \ \Phi_5$ for some $\Phi_5$. From $\Theta[w/x] \ \overline{\lhd}_S \ \Delta'$ and $\Delta' \ \overline{\lhd}_S \ \Phi_5$, we have $\Theta[w/x] \ \overline{\mathcal{R}} \ \Phi_5$. Putting it all together, we have:

$$\Phi \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_2 \xrightarrow{a(x)} \Phi_3, \qquad \Phi_3[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_5, \qquad \Theta[w/x] \ \overline{\mathcal{R}} \ \Phi_5.$$

Thus $\mathcal{R}$ is indeed a simulation. □

**Proposition 4.13.** The relation $\overline{\lhd}_{FS}$ is reflexive and transitive.

*Proof.* Reflexivity of $\overline{\lhd}_{FS}$ follows from Lemma 4.8. To show transivity, we use a similar argument as in the proof of Proposition 4.12: define $\mathcal{R}$ such that $s \ \mathcal{R} \ \Theta$ iff there exists $\Delta$ such that $s \ \lhd_{FS} \ \Delta$ and $\Delta \ \overline{\lhd}_{FS} \ \Theta$. We show that $\mathcal{R}$ is a failure simulation.

Suppose $s \ \mathcal{R} \ \Theta$. The matching up of transitions between $s$ and $\Theta$ is proved similarly to the case with simulation, by proving the analog of Lemmas 4.9 - 4.11 for failure simulation. It then remains to show that when $s \not\downarrow_X$ then there exists $\Theta'$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \not\downarrow_X$ . Since $s \ \mathcal{R} \ \Theta$, by the definition of $\mathcal{R}$, we have a $\Delta$ s.t. $s \ \lhd_{FS} \ \Delta$ and $\Delta \ \overline{\lhd}_{FS} \ \Theta$. The former implies that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta' \not\downarrow_X$, for some $\Delta'$. It can be shown that, using arguments similar to the proof of Lemma 4.11 that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ for some $\Theta'$ such that $\Delta' \ \overline{\lhd}_{FS} \Theta'$. Suppose $\lceil \Delta' \rceil = \{s_i\}_{i \in I}$, i.e., $\Delta' = \sum_{i \in I} p_i \cdot \delta[s_i]$ with $\sum_{i \in I} p_i = 1$. Obviously, $s_i \not\downarrow_X$ for each $i \in I$. By Proposition 4.2, $\Theta = \sum_{i \in I} p_i \cdot \Theta_i$ for some distributions $\Theta_i$ such that $\delta[s_i] \ \overline{\lhd}_{FS} \ \Theta_i$. The latter implies, by Definition 4.1, that $s_i \ \lhd_{FS} \ \Theta_i$. Since $s_i \not\downarrow_X$, it follows that $\Theta_i \stackrel{\hat{\tau}}{\Longrightarrow} \Theta_i' \not\downarrow_X$, for some $\Theta_i'$. Thus $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} (\sum_{i \in I} p_i \cdot \Theta_i) \not\downarrow_X$ . □

**Corollary 4.14.** The relations $\sqsubseteq_S$ and $\sqsubseteq_{FS}$ are preorders.

*Proof.* The fact that $\sqsubseteq_S$ is a preorder follows from Lemma 4.11 and Proposition 4.12. Similar arguments hold for $\sqsubseteq_{FS}$, using an analog of Lemma 4.11 and Proposition 4.13. □

## 5. Soundness of the simulation preorders

In proving soundness of the simulation preorders with respect to testing preorders, we first need to prove certain congruence properties, i.e., closure under restriction and parallel composition. For this, it is helpful to consider a slightly more general definition of simulation, which incorporates another relation. This technique, called the *up-to* technique, has been used in the literature to prove congruence properties of various pre-orders for the $\pi$-calculus [22].

**Definition 5.1 (Up-to rules).** Let $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$. Define the relation $\mathcal{R}^t$ where $t \in \{r, \nu, p\}$ as the smallest relation which satisfies the closure rule for $t$, given below (where $\sigma$ is a renaming substitution):

$$\frac{s\ \mathcal{R}\ \Delta}{s\sigma\ \mathcal{R}^r\ \Delta\sigma}\ r \qquad \frac{s\ \mathcal{R}\ \Delta}{(\nu\vec{x}.s)\ \mathcal{R}^\nu\ (\nu\vec{x}.\Delta)}\ \nu \qquad \frac{s_1\ \mathcal{R}\ \Delta_1 \quad s_2\ \mathcal{R}\ \Delta_2}{(s_1 \mid s_2)\ \mathcal{R}^p\ (\Delta_1 \mid \Delta_2)}\ p$$

**Definition 5.2 ((Failure) Simulation up-to).** A relation $\mathcal{R} \subseteq S_p \times \mathcal{D}(S_p)$ is said to be a *(failure) simulation up to renaming* (likewise, restriction and parallel composition) if it satisfies the clauses 1, and 2, (and 3 for failure simulation) in Definition 4.3, but with $\overline{\mathcal{R}}$ replaced by $\overline{\mathcal{R}^r}$ (respectively, $\overline{\mathcal{R}^\nu}$ and $\overline{\mathcal{R}^p}$).

It is easy to see that $\mathcal{R} \subseteq \mathcal{R}^t$ for any $t \in \{r, \nu\}$ (i.e., via the identity relation as renaming substitution in the former, and via the empty restriction in the latter). The following lemma is then an easy consequence.

**Lemma 5.3.** If $\mathcal{R}$ is a (failure) simulation then it is a (failure) simulation up-to renaming, and also a (failure) simulation up to restriction.

Our objective is really to show that simulation up-to parallel composition is itself a simulation. This would then entail that (the lifted) simulation is closed under parallel composition, from which soundness w.r.t. may-testing follows. We prove this indirectly in three stages:

- simulation up-to renaming is a simulation;
- simulation up-to restriction is a simulation up-to renaming (hence also a simulation by the previous item);
- and, finally, simulation up-to parallel composition is a simulation up-to restriction.

## 5.1. Up to renaming

Note that as a consequence of Lemma 4.5 (1), given an injective renaming substitution $\sigma$, we have: if $s\sigma \xrightarrow{\alpha'} \Delta'$ then there exists $\alpha$ and $\Delta$ such that $\alpha' = \alpha\sigma$, $\Delta' = \Delta\sigma$ and $s \xrightarrow{\alpha} \Delta$. This is proved by simply applying Lemma 4.5 (1) to $s\sigma \xrightarrow{\alpha'} \Delta'$ using the inverse of $\sigma$.

In the following, we shall write $\mathcal{R}^{tt}$ to denote $(\mathcal{R}^t)^t$, i.e., the result of applying the up-to closure rule $t$ twice to $\mathcal{R}$.

**Lemma 5.4.** $\mathcal{R}^{rr} = \mathcal{R}^r$.

**Lemma 5.5.** If $\Delta_1\ \overline{\mathcal{R}^r}\ \Delta_2$ then $(\Delta_1\sigma)\ \overline{\mathcal{R}^r}\ (\Delta_2\sigma)$ for any renaming substitution $\sigma$.

*Proof.* This follows from the fact that $\Delta_1\ \overline{\mathcal{R}^r}\ \Delta_2$ implies $\Delta_1\sigma\ \overline{\mathcal{R}^{rr}}\ \Delta_2\sigma$ and that $\mathcal{R}^{rr} = \mathcal{R}^r$. □

**Lemma 5.6.** If $\mathcal{R}$ is a (failure) simulation up to renaming, then $\mathcal{R}^r \subseteq \lhd_S$ (respectively, $\mathcal{R}^r \subseteq \lhd_{FS}$).

*Proof.* Suppose $\mathcal{R}$ is a simulation. It is enough to show that $\mathcal{R}^r$ is a simulation. So suppose $s\ \mathcal{R}^r\ \Delta$ and $s \xrightarrow{\alpha} \Theta$. By the definition of $\mathcal{R}^r$, $s = s'\sigma$ and $\Delta = \Delta'\sigma$ for some renaming substitution $\sigma$ and some $s'$ and $\Delta'$ such that $s'\ \mathcal{R}\ \Delta'$. There are several cases to consider depending on the type of $\alpha$.

- $\alpha$ is $\tau$ or a free action: By Lemma 4.5 (1) we have $s' \xrightarrow{\alpha'} \Theta'$ for some $\alpha'$ and $\Theta'$ such that $\alpha = \alpha'\sigma$ and $\Theta = \Theta'\sigma$. Since $\mathcal{R}$ is a simulation up to renaming, $s'\mathcal{R}\Delta'$ implies that $\Delta' \stackrel{\hat{\alpha'}}{\Longrightarrow} \Delta_1$ and $\Theta'\ \overline{\mathcal{R}^r}\ \Delta_1$. The former implies, by Lemma 4.5 (2), that $\Delta \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta_2$ for some $\Delta_2$ such that $\Delta_2 = \Delta_1\sigma$, while the latter implies, by Lemma 5.5, that $\Theta = (\Theta'\sigma)\ \overline{\mathcal{R}^r}\ (\Delta_1\sigma) = \Delta_2$.
- $\alpha = a(x)$ for some $a$ and $x$: In this case, $x \notin fn(s, \Delta)$, so we can assume, without loss of generality, that $x$ does not occur in $\sigma$. Using a similar argument as in the previous case, we have that $s' \xrightarrow{b(x)} \Theta'$ for some $b$ and $\Theta'$ such that $\sigma(b) = a$ and $\Theta = \Theta'\sigma$. Since $\mathcal{R}$ is a simulation up to renaming, $s'\mathcal{R}\Delta'$ implies that for every name $w$, there exist $\Delta^1_w$, $\Delta^2_w$ and $\Delta_w$ such that:

$$\Delta' \stackrel{\hat{\tau}}{\Longrightarrow} \Delta^1_w \xrightarrow{b(x)} \Delta^2_w, \qquad \Delta^2_w[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w, \quad \text{and} \tag{8}$$

$$\Theta'[w/x]\ \overline{\mathcal{R}^r}\ \Delta_w. \tag{9}$$

Let $\Phi_1 = \Delta^1_w\sigma$, $\Phi_2 = \Delta^2_w\sigma$ and $\Phi = \Delta_w\sigma$. From (8) and Lemma 4.5 (2) we get:

$$\Delta = \Delta'\sigma \stackrel{\hat{\tau}}{\Longrightarrow} \Delta^1_w\sigma = \Phi_1 \stackrel{a(x)}{\longrightarrow} \Delta^2_w\sigma = \Phi_2.$$

By (8), the freshness assumption of $x$ w.r.t. $\sigma$, and Lemma 4.5 (2), we get

$$\Phi_2[w/x] = \Delta^2_w\sigma[w/x] = \Delta^2_w[w/x]\sigma \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w\sigma = \Phi.$$

Finally, by (9) and Lemma 5.5, $\Theta[w/x] = \Theta'\sigma[w/x] = \Theta'[w/x]\sigma\ \overline{\mathcal{R}^r}\ \Delta_w\sigma = \Phi$.

- $\alpha = \bar{a}(x)$: This case can be proved similarly to the previous cases.

For the case where $\mathcal{R}$ is a failure simulation, we additionally need to show that whenever $s\ \mathcal{R}^r\ \Delta$ and $s \not\downarrow_X$, we have $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \not\downarrow_X$ for some $\Theta$. Since $s\mathcal{R}\Delta$, we have $s = s'\sigma$ and $\Delta = \Delta'\sigma$ for some $s'$, $\Delta$ and renaming substitution $\sigma$. Let $X' = X\sigma^{-1}$, i.e., $X'$ is the inverse image of $X$ under $\sigma$. Then we have that $s' \not\downarrow_{X'}$, and $\Delta' \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \not\downarrow_{X'}$ . Applying $\sigma^{-1}$ to the latter, we obtain $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \not\downarrow_X$ . □

**Lemma 5.7.** Suppose $P \sqsubseteq_S Q$ ($P \sqsubseteq_{FS} Q$) and $\sigma$ is a renaming substitution. Then $P\sigma \sqsubseteq_S Q\sigma$ (respectively, $P\sigma \sqsubseteq_{FS} Q\sigma$).

*Proof.* Immediate from Lemma 5.6. □

## 5.2. Up to name restriction

The following lemma says that transitions are closed under name restriction, if certain conditions are satisfied.

**Lemma 5.8.** 1. For every state-based process $s$, every action $\alpha$ and every list of names $\vec{x}$ such that $\{\vec{x}\} \cap n(\alpha) = \emptyset$, $s \stackrel{\alpha}{\longrightarrow} \Delta$ implies $\nu\vec{x}.s \stackrel{\alpha}{\longrightarrow} \nu\vec{x}.\Delta$.

2. For every $\Delta$ and $\Phi$, every action $\alpha$ and every list of names $\vec{x}$ such that $\{\vec{x}\} \cap n(\alpha) = \emptyset$, $\Delta \stackrel{\alpha}{\longrightarrow} \Phi$ implies $\nu\vec{x}.\Delta \stackrel{\alpha}{\longrightarrow} \nu\vec{x}.\Phi$.

3. Suppose $s \stackrel{\bar{a}b}{\longrightarrow} \Delta$ and suppose $\vec{x}$ and $\vec{y}$ are names such that $\{\vec{x}, \vec{y}\} \cap \{a, b\} = \emptyset$. Then $\nu\vec{x}\nu b\nu\vec{y}.s \stackrel{\bar{a}(b)}{\longrightarrow} \nu\vec{x}\nu\vec{y}.\Delta$.

**Lemma 5.9.** If $\Delta\ \overline{\mathcal{R}^\nu}\ \Theta$ then $(\nu\vec{x}.\Delta)\ \overline{\mathcal{R}^\nu}\ (\nu\vec{x}.\Theta)$

**Lemma 5.10.** If $\mathcal{R}$ is a (failure) simulation up to restriction, then $\mathcal{R}^\nu \subseteq \lhd_S$ (respectively, $\mathcal{R}^\nu \subseteq \lhd_{FS}$).

*Proof.* Suppose $\mathcal{R}$ is a simulation up to restriction. We show that $\mathcal{R}^\nu$ is a simulation up to renaming, hence by Lemma 5.6 we have $\mathcal{R}^\nu \subseteq \mathcal{R}^{\nu r} \subseteq \lhd_S$.

Suppose $s\ \mathcal{R}^\nu\Delta$ and $s \stackrel{\alpha}{\longrightarrow} \Theta$. By the definition of $\mathcal{R}^\nu$, we have that $s = \nu\vec{x}.s'$, $\Delta = \nu\vec{x}.\Delta'$, and $s'[\vec{y}/\vec{x}]\ \mathcal{R}\ \Delta'[\vec{y}/\vec{x}]$ for some $\vec{y}$ such that $\{\vec{y}\} \cap fn(s, \Delta) = \emptyset$.

There are several cases depending on how the transition $s \stackrel{\alpha}{\longrightarrow} \Theta$ is derived. Note that there may be implicit $\alpha$-renaming involved in the derivations of a transition judgment. We assume that the names $\vec{x}$ are chosen such that no $\alpha$-renaming is needed in deriving the transition relation $\nu\vec{x}.s' \stackrel{\alpha}{\longrightarrow} \Theta$, e.g., one such choice would be one that avoids clashes with the free names in $\vec{y}$, $s$, and $\Delta$.

- $\alpha$ is either $\tau$ or a free action. In this case, the transition must have been derived as follows:

$$\frac{\frac{s' \stackrel{\alpha}{\longrightarrow} \Theta'}{\vdots}}{\nu\vec{x}.s' \stackrel{\alpha}{\longrightarrow} \nu\vec{x}.\Theta'}\ res$$

where $\Theta = \nu\vec{x}.\Theta'$ and $n(\alpha) \cap \{\vec{x}\} = \emptyset$. Here a double-line in the inference rule indicates zero or more applications of the rule. An inspection on the operational semantics will reveal that in this case, $n(\alpha) \subseteq fn(s)$ and $fn(\Theta) \subseteq fn(s)$. So in particular, $\{\vec{y}\} \cap n(\alpha) = \emptyset$. We thus can apply the renaming substitution $[\vec{y}/\vec{x}, \vec{x}/\vec{y}]$ to get $s'[\vec{y}/\vec{x}] \stackrel{\alpha}{\longrightarrow} \Theta'[\vec{y}/\vec{x}]$. Since $s'[\vec{y}/\vec{x}]\ \mathcal{R}\ \Delta'[\vec{y}/\vec{x}]$, we have that $\Delta'[\vec{y}/\vec{x}] \stackrel{\alpha}{\Longrightarrow} \Delta''[\vec{y}/\vec{x}]$ and

$\Theta'[\vec{y}/\vec{x}]\ \overline{\mathcal{R}^\nu}\ \Delta''[\vec{y}/\vec{x}]$. The former implies, via Lemma 5.8 (1), that $\nu\vec{x}.\Delta' \stackrel{\alpha}{\Longrightarrow} \nu\vec{x}.\Delta''$ and the latter implies, via Lemma 5.9, that $(\nu\vec{x}.\Theta')\ \overline{\mathcal{R}^\nu}\ (\nu\vec{x}.\Delta'')$. Since $\mathcal{R}^\nu \subseteq (\mathcal{R}^\nu)^r$, we also have $(\nu\vec{x}.\Theta')\ \overline{\mathcal{R}^{\nu r}}\ (\nu\vec{x}.\Delta'')$.

- $\alpha = a(z)$: With a similar argument as in the previous case, we can show that in this case we must have $s \xrightarrow{a(z)} \Theta'$ where $\Theta = \nu\vec{x}.\Theta'$. We need to show that for every name $w$, there exist $\Gamma^1_w$, $\Gamma^2_w$ and $\Gamma_w$ such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma^1_w \xrightarrow{a(z)} \Gamma^2_w$, $\Gamma^2_w[w/z] \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma_w$, and $\Theta[w/z]\ \overline{\mathcal{R}^{\nu r}}\ \Gamma_w$.
  Note that $z \notin \{\vec{x}\}$, but it may be the case that $z \in \{\vec{y}\}$. So we first apply a renaming $[u/z, z/u, \vec{y}/\vec{x}, \vec{x}/\vec{y}]$, for some fresh name $u$, to the transition $s' \xrightarrow{a(z)} \Theta'$ to get $s'[\vec{y}/\vec{x}] \xrightarrow{a(u)} \Theta'[u/z, \vec{y}/\vec{x}]$.
  Since $s'[\vec{y}/\vec{x}]\ \mathcal{R}\ \Delta'[\vec{y}/\vec{x}]$, we have, for every name $w$, some $\Delta^1_w$, $\Delta^2_w$ and $\Delta_w$ such that

$$\Delta'[\vec{y}/\vec{x}] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta^1_w \xrightarrow{a(u)} \Delta^2_w, \qquad \Delta^2_w[w/u] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w, \qquad \text{and} \tag{10}$$

$$\Theta'[u/z, \vec{y}/\vec{x}][w/u] = \Theta'[w/z, \vec{y}/\vec{x}]\ \overline{\mathcal{R}^\nu}\ \Delta_w[w/u]. \tag{11}$$

  Let $\Phi^1_w$, $\Phi^2_w$ and $\Phi_w$ be distributions such that $\Delta^1_w = \Phi^1_w[\vec{y}/\vec{x}]$, $\Delta^2_w = \Phi^2_w[u/z, \vec{y}/\vec{x}]$, and $\Delta_w = \Phi_w[\vec{y}/\vec{x}]$. So in particular, $\Delta^2_w[w/u] = \Phi^2_w[w/z, \vec{y}/\vec{x}]$ and $\Delta_w[w/u] = \Phi_w[w/z, \vec{y}/\vec{x}]$. Then (10) can be rewritten as:

$$\Delta'[\vec{y}/\vec{x}] \stackrel{\hat{\tau}}{\Longrightarrow} \Phi^1_w[\vec{y}/\vec{x}] \xrightarrow{a(u)} \Phi^2_w[u/z, \vec{y}/\vec{x}] \qquad \Phi^2_w[w/z, \vec{y}/\vec{x}] \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_w[\vec{y}/\vec{x}], \tag{12}$$

  and (11) can be rewritten as:

$$\Theta'[w/z, \vec{y}/\vec{x}]\ \overline{\mathcal{R}^\nu}\ \Phi_w[w/z, \vec{y}/\vec{x}]. \tag{13}$$

  Now, to define $\Gamma^1_w$, $\Gamma^2_w$ and $\Gamma_w$, we need to consider two cases, based on the value of $w$. The reason is that in the construction of $\Gamma_w$ we need to bound the free names in $\Phi_w$, so if $z$ is substituted with a name in $\vec{y}$, it could get captured.

  - $w \notin \{\vec{x}, \vec{y}\}$. In this case, define:

$$\Gamma^1_w = \nu\vec{x}.\Phi^1_w, \qquad \Gamma^2_w = \nu\vec{x}.\Phi^2_w, \qquad \Gamma_w = \nu\vec{x}.\Phi_w.$$

    By Lemma 5.8 (1) and (12), we have:

$$\nu\vec{x}.\Delta' \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma^1_w \xrightarrow{a(z)} \Gamma^2_w, \qquad \Gamma^2_w[w/z] \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma_w$$

    and by Lemma 5.9 and (13), we have

$$(\Theta[w/z]) = (\nu\vec{x}.\Theta')[w/z]\ \overline{\mathcal{R}^\nu}\ \Gamma_w,$$

    hence also, $(\Theta[w/z]) = (\nu\vec{x}.\Theta')[w/z]\ \overline{\mathcal{R}^{\nu r}}\ \Gamma_w$.
  - $w \in \{\vec{x}, \vec{y}\}$. Let $v$ be a new name (distinct from all other names considered so far). From the previous case, we know how to construct $\Gamma^1_v$, $\Gamma^2_v$ and $\Gamma_v$ such that

$$\nu\vec{x}.\Delta' \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma^1_v \xrightarrow{a(z)} \Gamma^2_v, \qquad \Gamma^2_v[v/z] \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma_v \qquad (\Theta[v/z])\ \overline{\mathcal{R}^{\nu r}}\ \Gamma_v. \tag{14}$$

    In this case, let $\Gamma^1_w = \Gamma^1_v$, $\Gamma^2_w = \Gamma^2_v$ and $\Gamma_w = \Gamma_v[w/v]$. (Note that because subsitution is capture-avoiding, the bound names in $\Gamma_v$ will be renamed via $\alpha$-conversion). Then by Lemma 4.5 (2) and Lemma 5.5 and (14):

$$\nu\vec{x}.\Delta' \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma^1_w \xrightarrow{a(z)} \Gamma^2_w, \qquad \Gamma^2_v[w/z] \stackrel{\hat{\tau}}{\Longrightarrow} \Gamma_w \qquad (\Theta[w/z])\ \overline{\mathcal{R}^{\nu r}}\ \Gamma_w.$$

- If $\alpha$ is a bound output action, i.e., $\alpha = \bar{a}(b)$ for some $a$ and $b$. There are two subcases to consider, depending on whether $b \in \{\vec{x}\}$ (i.e., one of the restriction names $\vec{x}$ is extruded) or not. The latter can be proved similarly to the previous case. We show here a proof of the former case. So suppose $b \in \vec{x}$, i.e., $\nu\vec{x} = \nu\vec{x}_1\nu b\nu\vec{x}_2$ and suppose that $[\vec{y}/\vec{x}]$ maps $b$ to $c$, i.e., $\nu\vec{y} = \nu\vec{y}_1\nu c\nu\vec{y}_2$. Suppose the transition relation

is derived as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{s \xrightarrow{\bar{a}b} \Theta'}{}\ res}{\nu\vec{x}_2.s \xrightarrow{\bar{a}b} \nu\vec{x}_2.\Theta'}\ open}{\nu b\nu\vec{x}_2.s \xrightarrow{\bar{a}(b)} \nu\vec{x}_2.\Theta'}\ res}{\nu\vec{x}_1\nu b\nu\vec{x}_2.s' \xrightarrow{\bar{a}(b)} \nu\vec{x}_1\nu\vec{x}_2.\Theta'}$$

Applying the renaming $[\vec{y}/\vec{x}, \vec{x}/\vec{y}]$ we have: $s[\vec{y}/\vec{x}] \xrightarrow{\bar{a}c} \Theta'[\vec{y}/\vec{x}]$. Since $s'[\vec{y}/\vec{x}]\ \mathcal{R}\ \Delta'[\vec{y}/\vec{x}]$, we have that

$$\Delta'[\vec{y}/\vec{x}] \stackrel{\bar{a}c}{\Longrightarrow} \Phi, \qquad \text{and} \qquad \Theta'[\vec{y}/\vec{x}]\ \overline{\mathcal{R}^\nu}\ \Phi. \tag{15}$$

Let $\Psi[\vec{y}/\vec{x}] = \Phi$. Lemma 5.8 (3) and (15) imply that

$$\nu\vec{x}.\Delta' = \nu\vec{y}_1\nu c\vec{y}_2.\Delta'[\vec{y}/\vec{x}] \stackrel{\bar{a}(c)}{\Longrightarrow} \nu\vec{y}_1\nu\vec{y}_2.\Psi[\vec{y}/\vec{x}] = \nu\vec{x}_1\vec{x}_2.\Psi[c/b]$$

and by an application of a renaming (Lemma 4.5 (1)) we get $\nu\vec{x}.\Delta' \stackrel{\bar{a}(b)}{\Longrightarrow} \nu\vec{x}_1\nu\vec{x}_2.\Psi$. Lemma 5.9 and (15) imply $(\nu\vec{x}_1\nu\vec{x}_2.\Theta'[c/b])\ \overline{\mathcal{R}^\nu}\ (\nu\vec{x}_1\nu\vec{x}_2.\Psi[c/b])$ hence, via the renaming $[c/b, b/c]$, $(\nu\vec{x}_1\nu\vec{x}_2.\Theta')\ \overline{\mathcal{R}^{\nu\tau}}\ (\nu\vec{x}_1\nu\vec{x}_2.\Psi)$.

If $\mathcal{R}$ is a failure simulation up to restriction, we need to additionally show that $\mathcal{R}^\nu$ satisfies clause 3 of Definition 4.3. Suppose $s\ \mathcal{R}^\nu\ \Theta$. Then $s = \nu\vec{x}.s'$ and $\Theta = \nu\vec{x}.\Theta'$ for some $\vec{x}$, $s'$ and $\Theta'$ such that $s'\ \mathcal{R}\ \Theta'$. Suppose $s \not\downarrow_X$ . We need to show that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$ such that $\Delta \not\downarrow_X$ for some $\Delta$. Since name restriction hides visible actions, it can be shown that $s' \not\downarrow_{X\setminus\{\vec{x}\}}$ iff $\nu\vec{x}.s' \not\downarrow_X$ . So from $s'\ \mathcal{R}\ \Theta'$ we have that $\Theta' \stackrel{\hat{\tau}}{\Longrightarrow} \Delta' \not\downarrow_{X\setminus\{\vec{x}\}}$ . Let $\Delta = \nu\vec{x}.\Delta'$. Then by Lemma 5.8 (2), we have $\Theta = \nu\vec{x}.\Theta' \stackrel{\hat{\tau}}{\Longrightarrow} \nu\vec{x}.\Delta' = \Delta \not\downarrow_X$ . □

**Lemma 5.11.** If $P \sqsubseteq_S Q$ ($P \sqsubseteq_{FS} Q$) then $(\nu\vec{x}.P)\ \sqsubseteq_S (\nu\vec{x}.Q)$ (respectively, $(\nu\vec{x}.P) \sqsubseteq_{FS} (\nu\vec{x}.Q)$).

*Proof.* This is a simple corollary of Lemma 5.3 and Lemma 5.10. □

### 5.3. Up to parallel composition

The following lemma will be useful in proving the closure of simulation under parallel composition. It is independent of the underlying calculus, and is originally proved in [9].

**Lemma 5.12.** 1. $(\sum_{j\in J} p_j \cdot \Phi_j) \mid (\sum_{k\in K} q_k \cdot \Delta_k) = \sum_{j\in J}\sum_{k\in K}(p_j \cdot q_k) \cdot (\Phi_j \mid \Delta_k)$.

2. Suppose $\mathcal{R}, \mathcal{R}' \subseteq S_p \times \mathcal{D}(S_p)$ are two relations such that $s\mathcal{R}'\Delta$ whenever $s = s_1 \mid s_2$ and $\Delta = \Delta_1 \mid \Delta_2$ with $s_1\mathcal{R}\Delta_1$ and $s_2\mathcal{R}\Delta_2$. Then $\Phi_1\overline{\mathcal{R}}\Delta_1$ and $\Phi_2\overline{\mathcal{R}}\Delta_2$ imply $(\Phi_1 \mid \Phi_2)\overline{\mathcal{R}'}(\Delta_1 \mid \Delta_2)$.

We also need a slightly more general substitution lemma for transitions than the one given in Lemma 4.5 (1). In the following, we denote with $n(\theta)$ the set of all names appearing in the domain and range of $\theta$.

**Lemma 5.13.** For any substitution $\sigma$, the following hold:

1. If $s \xrightarrow{\alpha} \Delta$ and $bn(\alpha) \cap n(\sigma) = \emptyset$ then $s\sigma \xrightarrow{\alpha\sigma} \Delta\sigma$.
2. If $\Delta \stackrel{\hat{\alpha}}{\Longrightarrow} \Phi$ and $bn(\alpha) \cap n(\sigma) = \emptyset$ then $\Delta\sigma \stackrel{\hat{\alpha}\sigma}{\Longrightarrow} \Phi\sigma$.

The following lemma shows that transitions are closed under parallel composition, under suitable conditions.

**Lemma 5.14.** 1. If $s \xrightarrow{\alpha} \Delta$ and $fn(s') \cap bn(\alpha) = \emptyset$ then $s \mid s' \xrightarrow{\alpha} \Delta \mid \delta[s']$ and $s' \mid s \xrightarrow{\alpha} \delta[s'] \mid \Delta$.

2. If $\Phi \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta$, where $\alpha$ is either $\tau$, a free action or a bound output, and $fn(\Phi') \cap bn(\alpha) = \emptyset$ then $\Phi \mid \Phi' \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta \mid \Phi'$ and $\Phi' \mid \Phi \stackrel{\hat{\alpha}}{\Longrightarrow} \Phi' \mid \Delta$.
3. If $\Phi \xrightarrow{a(y)} \Phi'$ and $\Delta \xrightarrow{\bar{a}w} \Delta'$ then $\Phi \mid \Delta \xrightarrow{\tau} \Phi'[w/y] \mid \Delta'$.

4. If $\Phi \xrightarrow{a(y)} \Phi'$ and $\Delta \xrightarrow{\bar{a}(y)} \Delta'$ then $\Phi \mid \Delta \xrightarrow{\tau} \nu y.(\Phi' \mid \Delta')$.

**Lemma 5.15.** If $\mathcal{R}$ is a simulation, then $\mathcal{R}^p \subseteq \lhd_S$.

*Proof.* We show that $\mathcal{R}^p$ is a simulation up to restriction, and therefore, by Lemma 5.10, it is included in $\lhd_S$. So suppose $s \ \mathcal{R}^p \ \Delta$ and $s \xrightarrow{\alpha} \Theta$. By definition, we have $s = s_1 \mid s_2$ and $\Delta = \Delta_1 \mid \Delta_2$ such that $s_1 \ \mathcal{R} \ \Delta_1$ and $s_2 \ \mathcal{R} \ \Delta_2$. There are several cases to consider depending on the type of $\alpha$:

- $\alpha$ is a free output action. There can be two ways in which the transition $s \xrightarrow{\alpha} \Theta$ is derived. We show here one case; the other case is symmetric. So suppose the transition is derived as follows:

$$\frac{s_1 \xrightarrow{\alpha} \Theta'}{s_1 \mid s_2 \xrightarrow{\alpha} \Theta' \mid \delta[s_2]}\ par$$

  where $\Theta = \Theta' \mid \delta[s_2]$. Since $s_1 \ \mathcal{R} \ \Delta_1$, we have $\Delta_1 \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta_1'$ and $\Theta' \ \overline{\mathcal{R}} \ \Delta_1'$. The former implies, via Lemma 5.14 (2), that $\Delta_1 \mid \Delta_2 \stackrel{\hat{\alpha}}{\Longrightarrow} \Delta_1' \mid \Delta_2$. Since $s_2 \ \mathcal{R} \ \Delta_2$ by assumption, and therefore $\delta[s_2] \ \overline{\mathcal{R}} \ \Delta_2$, by Lemma 5.12 (2) we have $\Theta = (\Theta' \mid \delta[s_2]) \ \overline{\mathcal{R}^p} \ (\Delta_1' \mid \Delta_2)$ and therefore also $\Theta = (\Theta' \mid \delta[s_2]) \ \overline{\mathcal{R}^{p\nu}} \ (\Delta_1' \mid \Delta_2)$.
- $\alpha = a(y)$ and $y \notin fn(s, \Delta)$. That is, in this case, the transition is derived as follows:

$$\frac{s_1 \xrightarrow{a(y)} \Theta'}{s_1 \mid s_2 \xrightarrow{a(y)} \Theta' \mid \delta[s_2]}\ par$$

  and $y \notin fn(s_2)$. (There is another symmetric case which we omit here.) Since $s_1 \ \mathcal{R} \ \Delta_1$, we have, for every name $w$, some $\Delta_w^1$, $\Delta_w^2$ and $\Delta_w$ such that:

$$\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w^1 \xrightarrow{a(y)} \Delta_w^2, \qquad \Delta_w^2[w/y] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w, \quad \text{and} \tag{16}$$

$$\Theta'[w/y] \ \overline{\mathcal{R}} \ \Delta_w. \tag{17}$$

  From (16) above and Lemma 5.14 (2), and the assumption that $y \notin fn(s, \Delta)$, we have

$$\Delta_1 \mid \Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w^1 \mid \Delta_2 \xrightarrow{a(y)} \Delta_w^2 \mid \Delta_2, \qquad \Delta_w^2[w/y] \mid \Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_w \mid \Delta_2.$$

  Since $s_2 \ \mathcal{R} \ \Delta_2$, and therefore $\delta[s_2] \ \overline{\mathcal{R}} \ \Delta_2$, it then follows from (17) and Lemma 5.12 (2) that $\Theta[w/y] = (\Theta'[w/y] \mid \delta[s_2]) \ \overline{\mathcal{R}^p} \ (\Delta_w \mid \Delta_2)$ and therefore $\Theta[w/y] = (\Theta'[w/y] \mid \delta[s_2]) \ \overline{\mathcal{R}^{p\nu}} \ (\Delta_w \mid \Delta_2)$.
- $\alpha = \bar{a}(y)$ and $y \notin fn(s, \Delta)$. This case is similar to the previous cases, except that we only need to consider an instantiation of $y$ with a fresh name.
- $\alpha = \tau$ and the transition $s \xrightarrow{\tau} \Theta$ is derived via a **Com**-rule. We show here one case; the other case can be dealt with symmetrically. So suppose the transition is derived as follows:

$$\frac{s_1 \xrightarrow{a(y)} \Theta_1 \quad s_2 \xrightarrow{\bar{a}w} \Theta_2}{s_1 \mid s_2 \xrightarrow{\tau} \Theta_1[w/y] \mid \Theta_2}\ com$$

  Without loss of generality, we can assume that $y \notin fn(s, \Delta)$. Since $s_1 \ \mathcal{R} \ \Delta_1$ and $s_2 \ \mathcal{R} \ \Delta_2$, we have:
  - For every name $w$, there are $\Lambda_1$, $\Lambda_2$ and $\Delta_1^w$ such that

$$\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Lambda_1 \xrightarrow{a(y)} \Lambda_2, \qquad \Lambda_2[w/y] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1^w \qquad \text{and} \tag{18}$$

$$\Theta_1[w/y] \ \overline{\mathcal{R}} \ \Delta_1^w \tag{19}$$

  - There exists $\Delta_2'$ such that

$$\Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_1 \xrightarrow{\bar{a}w} \Phi_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_2' \qquad \text{and} \tag{20}$$

$$\Theta_2 \ \overline{\mathcal{R}} \ \Delta_2' \tag{21}$$

From (18), (20), and Lemma 5.14 (2)-(3), we have:

$$\Delta_1 \mid \Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Lambda_1 \mid \Phi_1 \stackrel{\tau}{\longrightarrow} \Lambda_2[w/y] \mid \Phi_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1^w \mid \Delta_2',$$

and Lemma 5.12 (2), together with (19) and (21), implies $(\Theta_1[w/y] \mid \Theta_2)\ \overline{\mathcal{R}^p}\ (\Delta_1^w \mid \Delta_2')$ and therefore $(\Theta_1[w/y] \mid \Theta_2)\ \overline{\mathcal{R}^{p\nu}}\ (\Delta_1^w \mid \Delta_2')$.

- $\alpha = \tau$ and the transition $s \stackrel{\tau}{\longrightarrow} \Theta$ is derived via the **Close**-rule:

$$\frac{s_1 \stackrel{a(y)}{\longrightarrow} \Theta_1 \quad s_2 \stackrel{\bar{a}(y)}{\longrightarrow} \Theta_2}{s_1 \mid s_2 \stackrel{\tau}{\longrightarrow} \nu y.(\Theta_1 \mid \Theta_2)}\ close.$$

Again, we only show one of the two symmetric cases. Without loss of generality, assume that $y$ is chosen to be fresh w.r.t. $s$ and $\Delta$. Since $s_1\ \mathcal{R}\Delta_1$ and $s_2\ \mathcal{R}\Delta_2$, we have:

– For every name $w$, there are $\Lambda_1$, $\Lambda_2$ and $\Delta_1^w$ such that

$$\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Lambda_1 \stackrel{a(y)}{\longrightarrow} \Lambda_2, \qquad \Lambda_2[w/y] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1^w \qquad \text{and} \qquad \Theta_1[w/y]\ \overline{\mathcal{R}}\ \Delta_1^w.$$

Note that letting $w = y$, we have

$$\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Lambda_1 \stackrel{a(y)}{\longrightarrow} \Lambda_2, \qquad \Lambda_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1^y \qquad \text{and} \tag{22}$$

$$\Theta_1\ \overline{\mathcal{R}}\ \Delta_1^y \tag{23}$$

– There exist $\Phi_1$, $\Phi_2$ and $\Delta_2'$ such that

$$\Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_1 \stackrel{\bar{a}(y)}{\longrightarrow} \Phi_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_2' \qquad \text{and} \tag{24}$$

$$\Theta_2\ \overline{\mathcal{R}}\ \Delta_2' \tag{25}$$

Then, by (22), (24), Lemma 5.14 (2) and (4), and Lemma 5.8 (1), we have:

$$\Delta_1 \mid \Delta_2 \stackrel{\hat{\tau}}{\Longrightarrow} \Lambda_1 \mid \Phi_1 \stackrel{\tau}{\longrightarrow} \nu y.(\Lambda_2 \mid \Phi_2) \stackrel{\hat{\tau}}{\Longrightarrow} \nu y.(\Delta_1^y \mid \Delta_2').$$

Lemma 5.12 (2), together with (23) and (25), implies $(\Theta_1 \mid \Theta_2)\ \overline{\mathcal{R}^p}\ (\Delta_1^y \mid \Delta_2')$, which also means: $(\Theta_1 \mid \Theta_2)\ \overline{\mathcal{R}^{p\nu}}\ (\Delta_1^y \mid \Delta_2')$. Now by Lemma 5.9, the latter implies that

$$\nu y.(\Theta_1 \mid \Theta_2)\ \overline{\mathcal{R}^{p\nu}}\ \nu y.(\Delta_1^y \mid \Delta_2').$$

□

**Lemma 5.16.** If $\mathcal{R}$ is a failure simulation, then $\mathcal{R}^p \subseteq \lhd_{FS}$.

*Proof.* Suppose $s\mathcal{R}^p\Delta$ and $s \not\downarrow_X$. By definition, we have $s = s_1 \mid s_2$ and $\Delta = \Delta_1 \mid \Delta_2$ such that $s_1\ \mathcal{R}\ \Delta_1$ and $s_2\ \mathcal{R}\ \Delta_2$. Then we have $s_i \not\downarrow_X$ for $i = 1, 2$. Define a set $A$ as follows:

$$A = \{a, \bar{a} \mid a \in fn(s_1, s_2, \Delta_1, \Delta_2)\} \cup X.$$

That is, $A$ contains the set of free (co-)names in $s_i$ and $\Delta_i$ and $X$. Let $X_i$ be the largest set such that $X \subseteq X_i \subseteq A$ and $s_i \not\downarrow_{X_i}$. Since $\mathcal{R}$ is a failure simulation, it follows that there exist $\Delta_i'$ such that $\Delta_i \stackrel{\tau}{\Longrightarrow} \Delta_i' \not\downarrow_{X_i}$. By Lemma 5.14 (2), we have $\Delta_1 \mid \Delta_2 \stackrel{\tau}{\Longrightarrow} \Delta_1' \mid \Delta_2'$. We claim that $(\Delta_1' \mid \Delta_2') \not\downarrow_X$. Suppose otherwise, that is, there exist $t_1 \in \lceil \Delta_1' \rceil$ and $t_2 \in \lceil \Delta_2' \rceil$ such that either $(t_1 \mid t_2) \downarrow_\mu$, for some $\mu \in X$, or $(t_1 \mid t_2) \stackrel{\tau}{\longrightarrow}$. If $(t_1 \mid t_2) \downarrow_\mu$ then our operational semantics entails that either $t_1 \downarrow_\mu$ or $t_2 \downarrow_\mu$, which contradicts the fact that $\Delta_i' \not\downarrow_{X_i}$. So let's assume that $(t_1 \mid t_2) \stackrel{\tau}{\longrightarrow}$. Again, from the assumption $\Delta_i' \not\downarrow_{X_i}$, we can immediately rule out the cases where $t_i \stackrel{\tau}{\longrightarrow}$ or $t_i \downarrow_\mu$, for some $\mu \in X$. This leaves us only with the cases where $t_1 \stackrel{\mu}{\longrightarrow}$ and $t_2 \stackrel{\bar{\mu}}{\longrightarrow}$ where $\mu \notin X$ and $\bar{\mu} \notin X$. But since $\Delta_i' \not\downarrow_{X_i}$, this can only be the case if $\mu \notin X_1$ and $\bar{\mu} \notin X_2$. From the operational semantics, it is easy to see that $fn(\Delta_1', \Delta_2') \subseteq fn(\Delta_1, \Delta_2)$, so it must be the case that $\mu \in A$

and $\bar{\mu} \in A$. It also must be the case that $s_1 \downarrow_\mu$, for otherwise, it would contradict the "largest" property of $X_1$. Similarly, we can argue that $s_2 \downarrow_{\bar{\mu}}$. But then this would imply that $(s_1 \mid s_2) \xrightarrow{\tau}$, contradicting the fact that $(s_1 \mid s_2) \not\downarrow_X$ .

The matching up of transitions and the using of $\mathcal{R}$ to prove the preservation property of $\lhd_{FS}$ under parallel composition are similar to those in the corresponding proof in Lemma 5.15 for simulations, so we omit them. □

**Lemma 5.17.** 1. If $P_1 \sqsubseteq_S Q_1$ and $P_2 \sqsubseteq_S Q_2$ then $P_1 \mid P_2 \ \sqsubseteq_S Q_1 \mid Q_2$.
2. If $P_1 \sqsubseteq_{FS} Q_1$ and $P_2 \sqsubseteq_{FS} Q_2$ then $P_1 \mid P_2 \ \sqsubseteq_{FS} Q_1 \mid Q_2$.

*Proof.* It is enough to show that $(\lhd_S)^p \subseteq \lhd_S$ and $(\lhd_{FS})^p \subseteq \lhd_{FS}$, which follow directly from Lemmas 5.15 and 5.16 respectively. □

## 5.4. Soundness

We now proceed to proving the main result, which is that $P \sqsubseteq_S Q$ implies $P \sqsubseteq_{pmay} Q$, and $P \sqsubseteq_{FS} Q$ implies $P \sqsubseteq_{pmust} Q$. The structure of the proof follows closely that of [7]. Most of the intermediate lemmas in this section are not specific to the π-calculus; rather, they utilise the underlying pLTS semantics.

Let $\pi^\omega$ be the set of all $\pi$ processes that may use action $\omega$. We write $s \xrightarrow{\alpha}_\omega \Delta$ if either $\alpha = \omega$ or $\alpha \neq \omega$ but both $s \not\xrightarrow{\omega}$ and $s \xrightarrow{\alpha} \Delta$ hold. We define $\xrightarrow{\hat{\tau}}_w$ as we did for $\xrightarrow{\hat{\tau}}$, using $\xrightarrow{\tau}_\omega$ in place of $\xrightarrow{\tau}$. Similarly, we define $\Longrightarrow_\omega$ and $\overset{\hat{\alpha}}{\Longrightarrow}_\omega$. Simulation and failure simulation are adapted to $\pi^\omega$ as follows.

**Definition 5.18.** Let $\lhd^e_{FS} \subseteq \pi^\omega \times \mathcal{D}(\pi^\omega)$ be the largest relation such that $s \lhd^e_{FS} \Theta$ implies

- If $s \xrightarrow{a(x)}_\omega \Delta$ and $x \notin fn(s, \Theta)$, then for every name $w$, there exists $\Theta_1$, $\Theta_2$ and $\Theta'$ such that
$$\Theta \overset{\hat{\tau}}{\Longrightarrow}_\omega \Theta_1 \xrightarrow{a(x)}_\omega \Theta_2, \qquad \Theta_2[w/x] \overset{\hat{\tau}}{\Longrightarrow}_\omega \Theta', \qquad \text{and} \qquad (\Delta[w/x]) \ \overline{\mathcal{R}} \ \Theta'.$$
- if $s \xrightarrow{\alpha}_\omega \Delta$ and $\alpha$ is not an input action, then there is some $\Theta'$ with $\Theta \overset{\hat{\alpha}}{\Longrightarrow}_\omega \Theta'$ and $\Delta \ \overline{\lhd^e_{FS}} \ \Theta'$
- if $s \not\downarrow_X$ with $\omega \in X$ then there is some $\Theta'$ with $\Theta \overset{\hat{\tau}}{\Longrightarrow}_\omega \Theta'$ and $\Theta' \not\downarrow_X$.

Similarly we can define $\lhd^e_S$ by dropping the third clause. Let $P \sqsubseteq^e_{FS} Q$ if $[\![P]\!] \overset{\hat{\tau}}{\Longrightarrow}_\omega \Theta$ for some $\Theta$ with $[\![Q]\!] \ \overline{\lhd^e_{FS}} \ \Theta$. Similarly, $P \sqsubseteq^e_S Q$ if $[\![Q]\!] \overset{\hat{\tau}}{\Longrightarrow}_\omega \Theta$ for some $\Theta$ with $[\![P]\!] \ \overline{\lhd^e_S} \ \Theta$.

Note that for $\pi$-processes $P, Q$, there is no action $\omega$, therefore we have $P \sqsubseteq_{FS} Q$ iff $P \sqsubseteq^e_{FS} Q$, and $P \sqsubseteq_S Q$ iff $P \sqsubseteq^e_S Q$.

**Lemma 5.19.** Let $P, Q$ be processes in $\pi$ and $T$ be a process in $\pi^\omega$.

1. If $P \sqsubseteq_S Q$ then $T \mid P \sqsubseteq^e_S T \mid Q$.
2. If $P \sqsubseteq_{FS} Q$ then $T \mid P \sqsubseteq^e_{FS} T \mid Q$.

*Proof.* Similar to the proof of Lemma 5.17. □

**Lemma 5.20.** 1. $P \sqsubseteq_{pmay} Q$ if and only if for every test $T$ we have
$$max(\mathbb{V}([\![\nu \vec{x}.(T \mid P)]\!])) \leq max(\mathbb{V}([\![\nu \vec{x}.(T \mid Q)]\!]))$$
where $\vec{x}$ contain the free names of $T$, $P$ and $Q$, excluding $\omega$.
2. $P \sqsubseteq_{pmust} Q$ if and only if for every test $T$ we have
$$min(\mathbb{V}([\![\nu \vec{x}.(T \mid P)]\!])) \leq min(\mathbb{V}([\![\nu \vec{x}.(T \mid Q)]\!]))$$
where $\vec{x}$ contain the free names of $T$, $P$ and $Q$, excluding $\omega$.

*Proof.* The results follow from the simple fact that, for non-empty finite outcome sets $O_1, O_2$,

- $O_1 \sqsubseteq_{Ho} O_2$ iff $max(O_1) \leq max(O_2)$

- $O_1 \sqsubseteq_{Sm} O_2$ iff $min(O_1) \leq min(O_2)$

which is established as Proposition 2.1 in [9]. □

**Lemma 5.21.** $\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_2$ implies $max(\mathbb{V}(\Delta_1)) \geq max(\mathbb{V}(\Delta_2))$ and $min(\mathbb{V}(\Delta_1)) \leq min(\mathbb{V}(\Delta_2))$.

*Proof.* Similar properties are proven in [9, Lemma 6.15] using a function *maxlive* instead of $max \circ \mathbb{V}$. Essentially the same arguments apply here. □

**Proposition 5.22.** 1. $\Delta_1 \overline{\lhd^e_S} \Delta_2$ implies $max(\mathbb{V}(\Delta_1)) \leq max(\mathbb{V}(\Delta_2))$.
2. $\Delta_1 \overline{\lhd^e_{FS}} \Delta_2$ implies $min(\mathbb{V}(\Delta_1)) \geq min(\mathbb{V}(\Delta_2))$.

*Proof.* The first clause is proven in [9, Proposition 6.16] using a function *maxlive* instead of $max \circ \mathbb{V}$. The second clause is proven in [7, Proposition 4.10] □

**Theorem 5.23.** 1. $P \sqsubseteq_S Q$ implies $P \sqsubseteq_{pmay} Q$
2. $P \sqsubseteq_{FS} Q$ implies $P \sqsubseteq_{pmust} Q$.

*Proof.* We prove the second statement; similar is the first one. Suppose $P \sqsubseteq_{FS} Q$. Given Proposition 5.20, it is sufficient to show that for every test $T$,

$$min(\mathbb{V}([\![\nu\vec{x}(T \mid P)]\!])) \leq min(\mathbb{V}([\![\nu\vec{x}(T \mid Q)]\!]))$$

where $\vec{x}$ contain the free names of $T$, $P$ and $Q$, but excluding $\omega$. Since $\sqsubseteq_{FS}$ is preserved by parallel composition (cf. Lemma 5.19) and name restriction, we have that $\nu\vec{x}(T \mid P) \sqsubseteq^e_{FS} \nu\vec{x}(T \mid Q)$, which means there is a $\Theta$ such that $[\![\nu\vec{x}(T \mid P)]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta$ and $[\![\nu\vec{x}(T \mid Q)]\!] \ \overline{\lhd^e_{FS}} \ \Theta$. The result then follows from Proposition 5.22 and Lemma 5.21. □

## 6. A modal logic for $\pi_p$

We consider a modal logic based on a fragment of Milner-Parrow-Walker's (MPW) modal logic for the (non-probabilistic) $\pi$-calculus [18], but extended with a probabilistic disjunction operator $\oplus$, similar to that used in [7]. The language of formulas is given by the following grammar:

$$\varphi ::= \top \ \mid \ \mathbf{ref}(X) \ \mid \ \langle a(x)\rangle\varphi \ \mid \ \langle \bar{a}x\rangle\varphi \ \mid \ \langle \bar{a}(x)\rangle\varphi \ \mid \ \varphi_1 \wedge \varphi_2 \ \mid \ \varphi_1 \,{}_p\!\oplus \varphi_2$$

The $x$'s in $\langle a(x)\rangle\varphi$ and $\langle \bar{a}(x)\rangle\varphi$ are binders, whose scope is over $\varphi$. The diamond operator $\langle a(x)\rangle$ is called a bound input modal operator, $\langle \bar{a}x\rangle$ a free output modal operator and $\langle \bar{a}(x)\rangle$ a bound output modal operator. In the formula $\mathbf{ref}(X)$, $X$ is a set consisting of names and/or co-names. Intuitively, $X$ in $\mathbf{ref}(X)$ can be seen as a "refusal" set, i.e., a process that satisfies this formula cannot make any transition on the channels in $X$. A process satisfies a probabilistic disjunction $\varphi_1 \,{}_p\!\oplus \varphi_2$ if it can be partitioned (via internal transitions) into two probabilistic distributions, one satisfying $\varphi_1$ and the other satisfying $\varphi_2$.

Instead of binary conjunction and probabilistic disjunction, we sometimes write $\bigwedge_{i\in I} \varphi_i$ and $\varphi_1 \,{}_p\!\oplus \varphi_2$ for finite index set $I$; they can be expressed by nested use of their binary forms. We refer to this modal logic as $\mathcal{F}$. Let $\mathcal{L}$ be the sub-logic of $\mathcal{F}$ by skipping the $\mathbf{ref}(X)$ clause. The semantics of $\mathcal{F}$ is defined as follows.

**Definition 6.1.** The *satisfaction relation* $\models$ between a distribution and a modal formula is defined inductively as follows:

- $\Delta \models \top$ always.
- $\Delta \models \mathbf{ref}(X)$ iff there is a $\Delta'$ with $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \not\downarrow_X$.
- $\Delta \models \langle a(x)\rangle\varphi$ iff for all $z$ there are $\Delta_1$, $\Delta_2$, $\Delta'$ and $w$ such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \stackrel{a(w)}{\longrightarrow} \Delta_2$, $\Delta_2[z/w] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \models \varphi[z/x]$.
- $\Delta \models \langle \bar{a}x\rangle\varphi$ iff for some $\Delta'$, $\Delta \stackrel{\widehat{\bar{a}x}}{\Longrightarrow} \Delta'$ and $\Delta' \models \varphi$.
- $\Delta \models \langle \bar{a}(x)\rangle\varphi$ iff for some $\Delta'$ and $w \notin fn(\varphi, \Delta)$, $\Delta \stackrel{\widehat{\bar{a}(w)}}{\Longrightarrow} \Delta'$ and $\Delta' \models \varphi[w/x]$.
- $\Delta \models \varphi_1 \wedge \varphi_2$ iff $\Delta \models \varphi_1$ and $\Delta \models \varphi_2$.

- $\Delta \models \varphi_1 \,{}_p\!\oplus \varphi_2$ iff there are $\Delta_1, \Delta_2 \in \mathcal{D}(S_p)$ with $\Delta_1 \models \varphi_1$ and $\Delta_2 \models \varphi_2$, such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} p \cdot \Delta_1 + (1-p) \cdot \Delta_2$.

We write $\Delta \sqsubseteq_{\mathcal{L}} \Theta$ just when $\Delta \models \psi$ implies $\Theta \models \psi$ for all $\psi \in \mathcal{L}$, and $\Delta \sqsubseteq_{\mathcal{F}} \Theta$ just when $\Theta \models \varphi$ implies $\Delta \models \varphi$ for all $\varphi \in \mathcal{F}$. We write $P \sqsubseteq_{\mathcal{L}} Q$ when $[\![P]\!] \sqsubseteq_{\mathcal{L}} [\![Q]\!]$, and $P \sqsubseteq_{\mathcal{F}} Q$ when $[\![P]\!] \sqsubseteq_{\mathcal{F}} [\![Q]\!]$.

Following [7], in order to show soundness of the logical preorders w.r.t. the simulation pre-orders, we need to define a notion of characteristic formulas.

**Definition 6.2 (Characteristic formula).** The *$\mathcal{F}$-characteristic formulas* $\varphi_s$ and $\varphi_\Delta$ of, respectively, a state-based process $s$ and a distribution $\Delta$ are defined inductively as follows:

$$\begin{array}{rcll} \varphi_s & := & \bigwedge\{\langle\alpha\rangle\varphi_\Delta \mid s \stackrel{\alpha}{\longrightarrow} \Delta\} \wedge \mathbf{ref}(\{\mu \mid s \not\Downarrow_\mu\}) & \text{if } s \stackrel{\tau}{\not\longrightarrow}, \\ \varphi_s & := & \bigwedge\{\langle\alpha\rangle\varphi_\Delta \mid s \stackrel{\alpha}{\longrightarrow} \Delta,\ \alpha \neq \tau\} \ \wedge\ \bigwedge\{\varphi_\Delta \mid s \stackrel{\tau}{\longrightarrow} \Delta\} & \text{otherwise.} \\ \varphi_\Delta & := & \bigoplus_{s \in \lceil\Delta\rceil} \Delta(s) \cdot \varphi_s & \end{array}$$

where $\bigoplus$ is a generalised probabilistic choice as in Section 2. The *$\mathcal{L}$-characteristic formulas* $\psi_s$ and $\psi_\Delta$ are defined likewise, but omitting the conjuncts $\mathbf{ref}(\{\mu \mid s \not\Downarrow_\mu\})$.

Note that because we use the late semantics (cf. Figure 1), the conjunction in $\varphi_s$ is finite even though there can be infinitely many (input) transitions from $s$.

**Example 6.3.** Let $P$ and $Q$ be as given in Example 4.4. The characteristic formula for $P$ is

$$\varphi_P = \langle a(x)\rangle(\langle\bar{a}c\rangle\top \,{}_{\frac{1}{2}}\!\oplus \top).$$

We show that $[\![Q]\!] \models \varphi_P$. By Definition 6.1, it is enough to show that, for every $z$, there are $\Delta_1$, $\Delta_2$ and $\Delta'$ such that $[\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \stackrel{a(x)}{\longrightarrow} \Delta_2$, $\Delta_2[z/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \models \langle\bar{a}c\rangle$. The first two conditions are satisfied by letting $\Delta_1 = [\![Q]\!]$, $\Delta_2 = \frac{1}{2} \cdot [x = b]\bar{a}c + \frac{1}{2} \cdot [x \neq b]\bar{a}c$ and $\Delta' = \Delta_2[z/x]$. It remains to show $\Delta' \models \langle\bar{a}c\rangle\top \,{}_{\frac{1}{2}}\!\oplus \top$. Let $\Delta'_1 = [z = b]\bar{a}c$ and $\Delta'_2 = [z \neq b]\bar{a}c$. Then obviously $\Delta' = \frac{1}{2} \cdot \Delta'_1 + \frac{1}{2} \cdot \Delta'_2$. To show $\Delta' \models \langle\bar{a}c\rangle\top \,{}_{\frac{1}{2}}\!\oplus \top$, we do case analysis on $z$:

- If $z = b$, then we have $\Delta'_1 \models \langle\bar{a}c\rangle\top$ and $\Delta'_2 \models \top$. Then $\Delta' \models \langle\bar{a}c\rangle\top \,{}_{\frac{1}{2}}\!\oplus \top$ follows immediately.
- Otherwise $z \neq b$. In this case, we have $\Delta'_1 \models \top$ and $\Delta'_2 \models \langle\bar{a}c\rangle\top$. Again, $\Delta' \models \langle\bar{a}c\rangle\top \,{}_{\frac{1}{2}}\!\oplus \top$ follows.

Given a state based process $s$, we define its *size*, $|s|$, as the number of process constructors and names in $s$. The following lemma is straightforward from the definition of the operational semantics of $\pi_p$.

**Lemma 6.4.** If $s \stackrel{\alpha}{\longrightarrow} \Delta$ then $|s| > |t|$ for every $t \in \lceil\Delta\rceil$.

**Lemma 6.5.** For every $\Delta \in \mathcal{D}(S_p)$, $\Delta \models \varphi_\Delta$, as well as $\Delta \models \psi_\Delta$.

*Proof.* It is enough to show that $\delta[s] \models \varphi_s$. This is proved by induction on $|s|$. So suppose $s \stackrel{\tau}{\not\longrightarrow}$. Then:

$$\begin{array}{rl} \varphi_s = & \mathbf{ref}(\{\mu \mid s \not\Downarrow_\mu\}) \wedge \\ & \bigwedge\{\langle a(x)\rangle\varphi_\Delta \mid s \stackrel{a(x)}{\longrightarrow} \Delta\} \wedge \bigwedge\{\varphi_\Delta \mid s \stackrel{\tau}{\longrightarrow} \Delta\} \wedge \\ & \bigwedge\{\langle\bar{a}x\rangle\varphi_\Delta \mid s \stackrel{\bar{a}x}{\longrightarrow} \Delta\} \wedge \bigwedge\{\langle\bar{a}(x)\rangle\varphi_\Delta \mid s \stackrel{\bar{a}(x)}{\longrightarrow} \Delta\}. \end{array}$$

where $\varphi_\Delta = \bigoplus_{s \in \lceil\Delta\rceil} \Delta(s).\varphi_s$. For each of the conjunct $\phi$, we prove that $\delta[s] \models \phi$. We show here two cases; the other cases are similar.

- $\phi = \mathbf{ref}(X)$, where $X = \{\mu \mid s \not\Downarrow_\mu\}$. For each $\mu \in X$ we have $s \not\Downarrow_\mu$. Since $s \stackrel{\tau}{\not\longrightarrow}$, we see that $s \not\Downarrow_X$.
- $\phi = \langle a(x)\rangle\varphi_\Delta$. So suppose $s \stackrel{a(x)}{\longrightarrow} \Delta$ and $\lceil\Delta\rceil = \{s_i \mid i \in I\}$ and $\Delta = \sum_{i \in I} p_i \cdot \delta[s_i]$. Since $|s_i| < |s|$, by the induction hypothesis, for every name $w$, we have $\delta[s_i[w/x]] \models \varphi_{s_i[w/x]}$ and therefore:

$$\Delta[w/x] = \sum_{i \in I} p_i \cdot \delta[s_i[w/x]] \models \bigoplus_{i \in I} p_i \cdot \varphi_{s_i[w/x]} = \varphi_\Delta[w/x].$$

Let $\Phi_1 = \Phi_2 = \delta[s]$. Obviously we have, for every $w$, $\Phi_1 \stackrel{\hat{\tau}}{\Longrightarrow} \Phi_2 \stackrel{a(x)}{\longrightarrow} \Delta$ and $\Delta[w/x] \models \varphi_\Delta[w/x]$. So by Definition 6.1, $\delta[s] \models \phi$.

□

**Lemma 6.6.** For any processes $P$ and $Q$, $[\![P]\!] \models \varphi_{[\![Q]\!]}$ implies $P \sqsubseteq_{FS} Q$, and likewise $[\![Q]\!] \models \psi_{[\![P]\!]}$ implies $P \sqsubseteq_S Q$.

*Proof.* Let $\mathcal{R}$ be the relation defined as follows: $s \mathcal{R} \Theta$ iff $\Theta \models \varphi_s$. We first prove the following claim:

$$\Theta \models \varphi_\Delta \text{ implies there exists } \Theta' \text{ such that } \Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \text{ and } \Delta \; \overline{\mathcal{R}} \; \Theta'. \tag{26}$$

To prove this claim (following [7]), suppose that $\Theta \models \Delta$. By definition, $\varphi_\Delta = \bigoplus_{i\in I} p_i \cdot \varphi_{s_i}$ and $\Delta = \sum_{i\in I} p_i \cdot \delta[s_i]$. For every $i \in I$, we have $\Theta_i \in \mathcal{D}(S_p)$ with $\Theta_i \models \varphi_{s_i}$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ with $\Theta' = \sum_{i\in I} p_i \cdot \Theta_i$. Since $s_i \; \mathcal{R} \; \Theta_i$ for all $i \in I$, we have $\Delta \; \overline{\mathcal{R}} \; \Theta'$.

We now proceed to show that $\mathcal{R}$ is a failure simulation, hence proving the first statement of the lemma. So suppose $s \; \mathcal{R} \; \Theta$.

1. Suppose $s \stackrel{\tau}{\longrightarrow} \Delta$. By the definition of $\mathcal{R}$, we have $\Theta \models \varphi_s$. By Definition 6.2, we also have $\Theta \models \varphi_\Delta$. By (26) above, there exists $\Theta'$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ and $\Delta \; \overline{\mathcal{R}} \; \Theta'$.
2. Suppose $s \stackrel{\bar{a}x}{\longrightarrow} \Delta$. Then by Definition 6.2, $\Theta \models \langle \bar{a}x \rangle \varphi_\Delta$. So $\Theta \stackrel{\bar{a}x}{\Longrightarrow} \Theta'$ and $\Theta' \models \varphi_\Delta$, for some $\Theta'$. By (26), there exists $\Theta''$ such that $\Theta' \stackrel{\hat{\tau}}{\Longrightarrow} \Theta''$ and $\Delta \; \overline{\mathcal{R}} \; \Theta''$. This means that $\Theta \stackrel{\bar{a}x}{\Longrightarrow} \Theta''$ and $\Delta \; \overline{\mathcal{R}} \; \Theta''$.
3. Suppose $s \stackrel{a(x)}{\longrightarrow} \Delta$ for some $x \notin fn(s, \Theta)$. By Definition 6.2, $\Theta \models \langle a(x) \rangle \varphi_\Delta$. This means for every name $z$, there exists $\Theta^1_z$, $\Theta^2_z$ and $\Theta_z$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta^1_z \stackrel{a(x)}{\longrightarrow} \Theta^2_z$, $\Theta^2_z[z/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta_z$ and $\Theta_z \models \varphi_\Delta[z/x]$.[1] Then by (26) we have $\Theta_z \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'_z$ and $\Delta[z/x] \; \overline{\mathcal{R}} \; \Theta'_z$. So we indeed have, for every name $z$, $\Theta^1_z$, $\Theta^2_z$ and $\Theta'_z$ s.t.
$$\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta^1_z \stackrel{a(x)}{\longrightarrow} \Theta^2_z, \qquad \Theta^2_z[z/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'_z \qquad \text{and} \qquad \Delta[z/x] \; \overline{\mathcal{R}} \; \Theta'_z.$$
4. Suppose $s \stackrel{\bar{a}(x)}{\longrightarrow} \Delta$. This case is similar to the previous one, except that we need only to consider one instance of $x$ with a fresh name.
5. Suppose $s \not\downarrow_X$ for a set of channel names $X$. By Definition 6.2, we have $\Theta \models \mathbf{ref}(X)$. Hence, there is some $\Theta'$ with $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ and $\Theta' \not\downarrow_X$.

To establish the second statement, define $\mathcal{R}$ by $s \mathcal{R} \Theta$ iff $\Theta \models \psi_s$. Just as above it can be shown that $\mathcal{R}$ is a simulation. Then the second statement of the lemma easily follows. □

**Theorem 6.7.** 1. If $P \sqsubseteq_{\mathcal{L}} Q$ then $P \sqsubseteq_S Q$.
2. If $P \sqsubseteq_{\mathcal{F}} Q$ then $P \sqsubseteq_{FS} Q$.

*Proof.* Suppose $P \sqsubseteq_{\mathcal{L}} Q$. By Lemma 6.5, we have $[\![P]\!] \models \psi_{[\![P]\!]}$, hence $[\![Q]\!] \models \psi_{[\![P]\!]}$. Then by Lemma 6.6, we have $P \sqsubseteq_S Q$. For the second statement, assume $P \sqsubseteq_{FS} Q$, we have $[\![Q]\!] \models \varphi_{[\![Q]\!]}$ and hence $[\![P]\!] \models \varphi_{[\![Q]\!]}$, and thus $P \sqsubseteq_{FS} Q$. □

## 7. Completeness of the simulation preorders

To prove completeness, we use the same approach as [7], by resorting to vector-based testing (see Section 3 and Theorem 3.3). In the following, we assume a function *new* that takes as an argument a finite set of names and outputs a fresh name, i.e., if $new(N) = x$ then $x \notin N$. If $N = \{x_1, \ldots, x_n\}$, we write $[x \neq N]P$ to abbreviate $[x \neq x_1][x \neq x_2] \cdots [x \neq x_n]P$.

[1] Strictly speaking, we should also consider the case where $\Theta^1_z \stackrel{a(w)}{\longrightarrow} \Theta^2_z$, but it is easy to see that since $x \notin fn(s, \Theta)$ we can always apply a renaming to rename $w$ to $x$.

For convenience of presentation, we write $\vec{\omega}$ for the vector in $[0,1]^\Omega$ defined by $\vec{\omega}(\omega) = 1$ and $\vec{\omega}(\omega') = 0$ for any $\omega' \neq \omega$. We also extend the $Apply^\Omega$ function to allow applying a test to a distribution, defined as $Apply^\Omega(T, \Delta) = \mathbb{V}(\nu\vec{x}([\![T]\!] \mid \Delta))$ where $\vec{x} = fn(T, \Delta) - \Omega$.

**Lemma 7.1.** If $\Delta \models \varphi$ then $\Delta\sigma \models \varphi\sigma$ for any renaming substitution $\sigma$.

In the following, given a name $a$, we write $a.P$ to denote $a(y).P$ for some $y \notin fn(P)$. Similarly, we write $\bar{a}.P$ to denote $\bar{a}a.P$. Recall that the size of a state-based process, $|s|$, is the number of symbols in $s$. The *size* of a distribution $\Delta$, written $|\Delta|$, is the *multiset* $\{|s| \mid s \in \lceil\Delta\rceil\}$. There is a well-founded ordering on $|\Delta|$, i.e., the multiset (of natural numbers) ordering, which we shall denote with $\prec$.

**Lemma 7.2.** Let $P$ be a process and $T, T_i$ be tests.

1. $o \in Apply^\Omega(\omega, P)$ iff $o = \vec{\omega}$.
2. Let $X = \{\mu_1, ..., \mu_n\}$ and $T = \mu_1.\omega + ... + \mu_n.\omega$. Then $\vec{0} \in Apply^\Omega(T, P)$ iff $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$ for some $\Delta$ with $\Delta \not\downarrow_X$.
3. Suppose the action $\omega$ does not occur in the test $T$. Then $o \in Apply^\Omega(\omega + a(x).([x = y]\tau.T + \omega), P)$ with $o(\omega) = 0$ iff there is $\Delta$ such that $[\![P]\!] \stackrel{\widehat{ay}}{\Longrightarrow} \Delta$ and $o \in Apply^\Omega(T[y/x], \Delta)$.
4. Suppose the action $\omega$ does not occur in the test $T$ and $fn(P) \subseteq N$. Then $o \in Apply^\Omega(\omega + a(x).([x \neq N]\tau.T + \omega), P)$ with $o(\omega) = 0$ iff there is $\Delta$ such that $[\![P]\!] \stackrel{\widehat{a(y)}}{\Longrightarrow} \Delta$ and $o \in Apply^\Omega(T[y/x], \Delta)$.
5. Suppose the action $\omega$ does not occur in the test $T$. Then $o \in Apply^\Omega(\omega + \bar{a}x.T, P)$ with $o(\omega) = 0$ iff there are $\Delta$, $\Delta_1$ and $\Delta_2$ such that $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \stackrel{a(y)}{\longrightarrow} \Delta_2$, $\Delta_2[x/y] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$ and $o \in Apply^\Omega(T, \Delta)$.
6. $o \in Apply^\Omega(\bigoplus_{i\in I} p_i \cdot T_i, P)$ iff $o = \sum_{i\in I} p_i \cdot o_i$ for some $o_i \in Apply^\Omega(T_i, P)$ for all $i \in I$.
7. $o \in Apply^\Omega(\sum_{i\in I} \tau.T_i, P)$ if for all $i \in I$ there are $q_i \in [0,1]$ and $\Delta_i$ such that $\sum_{i\in I} q_i = 1$, $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \sum_{i\in I} q_i \cdot \Delta_i$ and $o = \sum_{i\in I} q_i \cdot o_i$ for some $o_i \in Apply^\Omega(T_i, \Delta_i)$.

*Proof.* The proofs of items 1 and 2 are similar to the proofs of Lemma 6.7(1) and 6.7(2) in [7] for pCSP; items 6 and 7 correspond to Lemma 6.7(4) and Lemma 6.7(5) in [7], respectively. Items 3, 4 and 5 have a counterpart in Lemma 6.7(3) of [7], but they are quite different, due to the name-passing feature of the π-calculus, and the possibility of checking the identity of the input value via the match and the mismatch operators. We show here a proof of item 3; the proofs of items 4 and 5 are similar.

We first generalize item 3 to distributions: given $\omega$ and $T$ as above, we have, for every distribution $\Theta$,

$o \in Apply^\Omega(\omega + a(x).([x = y]\tau.T + \omega), \Theta)$ with $o(\omega) = 0$ iff there is $\Delta$ such that $\Theta \stackrel{\widehat{ay}}{\Longrightarrow} \Delta$ and $o \in Apply^\Omega(T[y/x], \Delta)$.

The 'if' part is straightforward from Definition 3.2. We show the 'only if' part here. The proof will make use of the following claim (easily proved by induction on $|\Theta|$):

$$\begin{array}{l}\textbf{Claim: } o \in Apply^\Omega([y = y]\tau.T[y/x] + \omega, \Theta) \text{ with } o(\omega) = 0 \text{ iff} \\ \text{there is } \Delta \text{ such that } \Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta \text{ and } o \in Apply^\Omega(T[y/x], \Delta).\end{array} \tag{27}$$

So, suppose we have $o \in Apply^\Omega(\omega + a(x).([x = y]\tau.T + \omega), \Theta)$ with $o(\omega) = 0$. We show, by induction on $|\Theta|$, that there exists $\Delta$ such that $\Theta \stackrel{\bar{a}y}{\Longrightarrow} \Delta$ and $o \in Apply^\Omega(T[y/x], \Delta)$. Let $T' = \omega + a(x).([x = y]\tau.T + \omega)$, and suppose $\Theta = p_1 \cdot \delta[s_1] + \ldots + p_n \cdot \delta[s_n]$, for pairwise distinct state-based processes $s_1, \ldots, s_n$, and suppose that $\vec{z}$ is an enumeration of the set $fn(T', \Theta) - \Omega$. Then

$$Apply^\Omega(T', \Theta) = \mathbb{V}^\Omega(p_1 \cdot \delta[\nu\vec{z}(T'|s_1)] + \ldots + p_n \cdot \delta[\nu\vec{z}(T'|s_n)]).$$

From Definition 3.2, in order to have $o(\omega) = 0$, it must be the case that $\nu\vec{z}(T'|s_j) \stackrel{\tau}{\longrightarrow}$ for every $j \in \{1, \ldots, n\}$. From the definition of the operational semantics, there are two cases where this might happen:

- For some $i$, $s_i \stackrel{\tau}{\longrightarrow} \Lambda$ for some distribution $\Lambda$. Let $\Theta' = p_1 \cdot \delta[s_1] + \ldots + p_i \cdot \Lambda + \ldots + p_n \cdot \delta[s_n]$. Then we have $\Theta \stackrel{\hat{\tau}}{\longrightarrow} \Theta'$ and $\nu\vec{z}(T'|\Theta) \stackrel{\hat{\tau}}{\longrightarrow} \nu\vec{z}(T'|\Theta')$. The latter means that $o \in \mathbb{V}^\Omega(\nu\vec{z}(T'|\Theta'))$ as well. By Lemma 6.4, we know that $|\Lambda| \prec \{|s_i|\}$, and therefore $|\Theta'| \prec |\Theta|$. By the induction hypothesis,

$$\Theta \stackrel{\hat{\tau}}{\longrightarrow} \Theta' \stackrel{\widehat{ay}}{\Longrightarrow} \Delta \qquad \text{and} \qquad o \in Apply^\Omega(T[y/x], \Delta).$$

- For every $i \in \{1,\ldots,n\}$, we have $s_i \stackrel{\tau}{\not\longrightarrow}$. This can only mean that the $\tau$ transition from $\nu\vec{z}(T'|s_i)$ derives from a communiation between $T'$ and $s_i$. This means that $s_i \downarrow_{\bar{a}}$, for every $i \in \{1,\ldots,n\}$. We claim that, in fact, for every $i$, we have $s_i \stackrel{\bar{a}y}{\longrightarrow} \Theta_i$, for some $\Theta_i$. For otherwise, we would have that for some $j$, $\nu\vec{z}(T'|s_j) \stackrel{\tau}{\longrightarrow} \nu\vec{z}(([u=y]\tau.T[y/x]+\omega) \mid \Theta_j)$, for some $u$ distinct from $y$. But this means that only the $\omega$ action is enabled in the test, so all results of $\mathbb{V}^\Omega(\nu\vec{z}(([u=y]\tau.T[y/x]+\omega) \mid \Theta_i))$ in this case would have a non-zero $\omega$ component, which would mean that $o(\omega)$ would be non-zero as well, contradicting the assumption that $o(\omega)=0$. So, we have $s_i \stackrel{\bar{a}y}{\longrightarrow} \Theta_i$ for every $i \in \{1,\ldots,n\}$. Let $\Theta' = p_1\cdot\Theta_1 + \ldots + p_n\cdot\Theta_n$. Then we have $\Theta \stackrel{\bar{a}y}{\longrightarrow} \Theta'$ and $\nu\vec{z}(T' \mid \Theta) \stackrel{\tau}{\longrightarrow} \nu\vec{z}(T'' \mid \Theta')$ where $T'' = [y=y]\tau.T[y/x]+\omega$. The latter transition means that $o \in \mathbb{V}^\Omega(\nu\vec{z}(T'' \mid \Theta')) = \mathit{Apply}^\Omega(T'',\Theta')$. We can therefore apply Claim 27 to get:

$$\Theta \stackrel{\bar{a}y}{\longrightarrow} \Theta' \stackrel{\hat{\tau}}{\Longrightarrow} \Delta \qquad \text{and} \qquad o \in \mathit{Apply}^\Omega(T[y/x],\Delta).$$

□

**Lemma 7.3.** If $o \in \mathit{Apply}^\Omega(\sum_{i\in I}\tau.T_i, P)$ then for all $i \in I$ there are $q_i \in [0,1]$ and $\Delta_i$ with $\sum_{i\in I} q_i = 1$ such that $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \sum_{i\in I} q_i\cdot\Delta_i$ and $o = \sum_{i\in I} q_i\cdot o_i$ for some $o_i \in \mathit{Apply}^\Omega(T_i,\Delta_i)$.

*Proof.* The proof is similar to the proof of Lemma 6.8 in [7]. □

The key to the completeness proof is to find a 'characteristic test' for every formula $\varphi \in \mathcal{L}$ with a certain property. The construction of these characteristic tests is given in the following lemma. Note that unlike in the case of pCSP [7], this construction is parameterised by a finite set of names $N$, representing the set of free names of the process/distribution on which the test applies to. This parameter is important for the test to be able to detect output of fresh names.

**Lemma 7.4.** For every finite set of names $N$ and every $\varphi \in \mathcal{F}$ such that $fn(\varphi) \subseteq N$, there exists a test $T_{\langle N,\varphi\rangle}$ and $v_\varphi \in [0,1]^\Omega$, such that

$$\Delta \models \varphi \qquad \text{iff} \qquad \exists o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle},\Delta) : o \le v_\varphi \tag{28}$$

for every $\Delta$ with $fn(\Delta) \subseteq N$, and in case $\varphi \in \mathcal{L}$ we also have

$$\Delta \models \varphi \qquad \text{iff} \qquad \exists o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle},\Delta) : o \ge v_\varphi. \tag{29}$$

$T_{\langle N,\varphi\rangle}$ is called a *characteristic test* of $\varphi$ and $v_\varphi$ its *target value*.

*Proof.* The characteristic tests and target values are defined by induction on $\varphi$:

- $\varphi = \top$: Let $T_{\langle N,\varphi\rangle} := \omega$ for some $\omega \in \Omega$ and $v_\varphi := \vec{\omega}$.
- $\varphi = \mathbf{ref}(X)$ with $X = \{\mu_1,...,\mu_n\}$. Let $T_\varphi := \mu_1.\omega + ... + \mu_n.\omega$ for some $\omega \in \Omega$, and $v_\varphi = \vec{0}$.
- $\varphi = \langle\bar{a}x\rangle\psi$: Let $T_{\langle N,\varphi\rangle} := \omega + a(y).([y=x]\tau.T_{\langle N,\psi\rangle} + \omega)$ for some $y \notin fn(T_{\langle N,\psi\rangle})$, where $\omega \in \Omega$ does not occur in $T_{\langle N,\psi\rangle}$ and $v_\varphi := v_\psi$.
- $\varphi = \langle\bar{a}(x)\rangle\psi$: Let $z = new(N)$ and $N' = N \cup \{z\}$. Without loss of generality, we can assume that $x = z$ (since we consider terms equivalent modulo $\alpha$-conversion). Then let $T_{\langle N,\varphi\rangle} := \omega + a(x).([x \neq N]\tau.T_{\langle N',\psi\rangle} + \omega)$, where $\omega \in \Omega$ does not occur in $T_{\langle N',\psi\rangle}$ and $v_\varphi := v_\psi$.
- $\varphi = \langle a(x)\rangle\psi$: Let $z = new(N)$ and $N' = N \cup \{z\}$. Let $p_w \in (0,1]$ for $w \in N'$ be chosen arbitrarily such that $\sum_{w\in N'} p_w = 1$. Then let

$$T_{\langle N,\varphi\rangle} := \bigoplus_{w\in N'} p_w\cdot(\omega_w + \bar{a}w.T_{\langle N',\psi[w/x]\rangle})$$

  where $\omega_w$ does not occur in $T_{\langle N',\psi[w/x]\rangle}$ for each $w \in N'$, and $\omega_{w_1} \neq \omega_{w_2}$ if $w_1 \neq w_2$. We let $v_\varphi := \sum_{w\in N'} p_w\cdot v_{\psi[w/x]}$.
- $\varphi = \bigwedge_{i\in I}\varphi_i$ where $I$ is a finite and non-empty index set. Choose an $\Omega$-disjoint family $(T_{\langle N,\varphi_i\rangle}, v_{\varphi_i})_{i\in I}$ of characteristic tests and target values. Let $p_i \in (0,1]$ for $i \in I$ be chosen arbitrarily s.t. $\sum_{i\in I} p_i = 1$.

Then let

$$T_{\langle N,\varphi\rangle} := \bigoplus_{i\in I} p_i \cdot T_{\langle N,\varphi_i\rangle} \qquad \text{and} \qquad v_\varphi := \sum_{i\in I} p_i \cdot v_{\varphi_i}.$$

- $\varphi = \bigoplus_{i\in I} p_i \cdot \varphi_i$. Choose an $\Omega$-disjoint family $(T_i, v_i)_{i\in I}$ of characteristic tests $T_i$ with target values $v_i$ for each $\varphi_i$, such that there are distinct success actions $\omega_i$ for $i \in I$ that do not occur in any of those tests. Let $T_i' := T_i \,{}_{\frac{1}{2}}\!\oplus \omega_i$ and $v_i' := \frac{1}{2}v_i + \frac{1}{2}\vec{\omega_i}$. Note that for all $i \in I$ also $T_i'$ is a characteristic test of $\varphi_i$ with target value $v_i'$. Let $T_{\langle N,\varphi\rangle} := \sum_{i\in I} \tau.T_{\langle N,\varphi_i\rangle}$ and $v_\varphi := \sum_{i\in I} p_i \cdot v_i'$.

  We now prove (28) above by induction on $\varphi$:

- $\varphi = \top$: obvious.
- $\varphi = \mathbf{ref}(X)$. Suppose $\Delta \models \varphi$. Then there is a $\Delta'$ with $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \not\downarrow_X$. By Lemma 7.2(2), $\vec{0} \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta)$.
  Now suppose $\exists o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta) : o \le v_\varphi$. This means $o = \vec{0}$, so by Lemma 7.2(2) there is a $\Delta'$ with $\Delta \stackrel{\tau}{\Longrightarrow} \Delta'$ and $\Delta' \not\downarrow_X$. Hence $\Delta \models \varphi$.
- $\varphi = \langle \bar{a}x\rangle\phi$ : Suppose $\Delta \models \varphi$. Then $\Delta \stackrel{\bar{a}x}{\Longrightarrow} \Delta'$ and $\Delta' \models \phi$. By the induction hypothesis, $\exists o \in \mathit{Apply}^\Omega(T_{\langle N,\phi\rangle}, \Delta') : o \le v_\phi$. By Lemma 7.2(3), this means $o \in \mathit{Apply}^\Omega(\omega + a(y).([y = x]\tau.T_{\langle N,\phi\rangle} + \omega), \Delta)$. Therefore, we have $o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta)$ and $o \le v_\varphi$.
  Conversely, suppose $\exists o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta) : o \le v_\varphi$. This implies $o(\omega) = 0$. By Lemma 7.2(3), this means $\Delta \stackrel{\bar{a}y}{\Longrightarrow} \Delta'$ and $o \in \mathit{Apply}^\Omega(T_{\langle N,\phi\rangle}, \Delta')$. By the induction hypothesis, we have $\Delta' \models \phi$, and therefore, by Definition 6.1, $\Delta \models \varphi$.
- $\varphi = \langle \bar{a}(x)\rangle\phi$ : This is similar to the previous case. The only difference is that the guard $[x \neq N]$ makes sure that it is the bound output transition that is enabled from $\Delta$, so we use Lemma 7.2(4) in place of Lemma 7.2(3).
- $\varphi = \langle a(x)\rangle\phi$ : Suppose $\Delta \models \varphi$. Then for every name $w$, there exist $\Delta_1$, $\Delta_2$ and $\Delta'$ such that:

$$\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \xrightarrow{a(x)} \Delta_2, \qquad \Delta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta', \qquad \text{and } \Delta' \models \phi[w/x]. \tag{30}$$

  In particular, (30) holds for any $w \in N'$, where $N' = N \cup \{new(N)\}$. By the induction hypothesis, $\exists o_w \in \mathit{Apply}^\Omega(T_{\langle N',\phi[w/x]\rangle}) : o_w \le v_{\langle N',\phi[w/x]\rangle}$, hence by Lemma 7.2(5),

$$o_w \in \mathit{Apply}^\Omega(\omega + \bar{a}w.T_{\langle N',\phi[w/x]\rangle}, \Delta)$$

  for each $w \in N'$. Then by Lemma 7.2(6), we have $o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta))$ where $o = \sum_{w\in N'} p_w \cdot o_w \le o_\varphi$. Suppose $\exists o \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta) : o \le v_\varphi$. Then by Lemma 7.2(6), we have $o = \sum_{w\in N'} p_w \cdot o_w$ for some $o_w$ with $o_w \in \mathit{Apply}^\Omega(\omega + \bar{a}w.T_{\langle N',\phi[w/x]\rangle}, \Delta)$. The latter means, by Lemma 7.2(5), for each $w \in N'$, there are $\Delta_1$, $\Delta_2$ and $\Delta'$ such that

$$\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_1 \xrightarrow{a(x)} \Delta_2, \qquad \Delta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta', \text{ and} \tag{31}$$

$$o_w \in \mathit{Apply}^\Omega(T_{\langle N',\phi[w/x]\rangle}, \Delta'). \tag{32}$$

  Since $\sum_{w\in N'} p_w \cdot o_w = o \le v_\varphi = \sum_{w\in N'} p_w \cdot v_{\phi[w/x]}$, we have

$$o_w \le v_{\phi[w/x]} \tag{33}$$

  for each $w \in N'$. Otherwise, suppose $o_w(\omega) > v_{\phi[w/x]}(\omega)$ for some $\omega \in \Omega$. We would have $o(\omega) = p_w \cdot o_w(\omega) > p_w \cdot v_{\phi[w/x]}(\omega) = v_\varphi(w)$, a contradiction to $o \le v_\varphi$. By (32), (33), and the induction hypothesis, we have

$$\Delta' \models \phi[w/x]. \tag{34}$$

  To show $\Delta \models \varphi$, we need to show for every $w$, there exist $\Delta_1$, $\Delta_2$ and $\Delta'$ satisfying (31) and (34) above. We have shown this for $w \in N'$. For the case where $w \notin N'$, this is obtained from the case where $x = z$ via the renaming $[w/z]$: Recall that $z \notin N$, so $z \notin fn(\Delta_2)$ and $z \notin fn(\phi)$. Therefore, we have, from

(31) and Lemma 4.5 (2), $\Delta_2[z/x][w/z] = \Delta_2[w/x] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'[w/z]$ and from (34) and Lemma 7.1, we have $\Delta'[w/z] \models \phi[w/x] = \phi[z/x][w/z]$.

- $\varphi = \bigwedge_{i\in I} \varphi_i$ : Suppose $\Delta \models \varphi$. Then $\Delta \models \phi_i$ for all $i \in I$, and by the induction hypothesis, $o_i \in \mathit{Apply}^\Omega(T_{\langle N,\phi_i\rangle}, \Delta) : o_i \leq v_{\varphi_i}$ and by Lemma 7.2(6)

$$\sum_{i\in I} p_i \cdot o_i \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta) \qquad \text{and} \qquad \sum_{i\in I} p_i \cdot o_i \leq \sum_{i\in I} p_i \cdot v_{\varphi_i} = v_\varphi.$$

  Suppose $\exists o \in \mathit{Apply}(T_{\langle N,\varphi\rangle}, \Delta) : o \leq v_\varphi$ Then by Lemma 7.2(6), $o = \sum_{i\in I} p_i \cdot o_i$ with $o_i \in \mathit{Apply}(T_{\langle N,\phi_i\rangle}, \Delta)$ for each $i \in I$. As in the last case, we see from $\sum_{i\in I} p_i \cdot o_i \leq \sum_{i\in I} p_i \cdot v_{\varphi_i}$ that $o_i \leq v_{\varphi_i}$ for each $i \in I$. By induction, we have $\Delta \models \phi_i$, therefore, by Definition 6.1, $\Delta \models \varphi$.

- $\varphi = \bigoplus_{i\in I} p_i \cdot \varphi_i$ : Suppose $\Delta \models \varphi$. Then $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \sum_{i\in I} p_i \cdot \Delta_i$ and $\Delta_i \models \phi_i$. By the induction hypothesis,

$$\exists o_i \in \mathit{Apply}^\Omega(T_i, \Delta_i) : o_i \leq v_i.$$

  Hence, there are $o'_i \in \mathit{Apply}^\Omega(T'_i, \Delta_i)$ with $o'_i \leq v'_i$. Thus by Lemma 7.2(7), $o := \sum_{i\in I} p_i \cdot o'_i \in \mathit{Apply}^\Omega(T_{\langle N,\varphi\rangle}, \Delta)$, and $o \leq v_\varphi$.
  Conversely, suppose $\exists o \in \mathit{Apply}(T_{\langle N,\varphi\rangle}, \Delta) : o \leq v_\varphi$. Then by Lemma 7.3, there are $q_i$ and $\Delta_i$, for all $i \in I$, such that $\sum_{i\in I} q_i = 1$ and $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \sum_{i\in I} q_i \cdot \Delta_i$ and $o = \sum_{i\in I} q_i \cdot o'_i$ for some $o'_i \in \mathit{Apply}^\Omega(T'_i, \Delta_i)$. Now $o'_i(\omega_i) = v'_i(\omega_i) = \frac{1}{2}$ for each $i \in I$. Using that $(T_i)_{i\in I}$ is an $\Omega$-disjoint family of tests, $\frac{1}{2}q_i = q_i o'_i(\omega_i) = o(\omega_i) \leq v_\varphi(\omega_i) = p_i v'_i(\omega_i) = \frac{1}{2}p_i$. As $\sum_{i\in I} q_i = \sum_{i\in I} p_i = 1$, it must be that $q_i = p_i$ for all $i \in I$. Exactly as in the previous case we obtain $o'_i \leq v'_i$ for all $i \in I$. Given that $T'_i = T_i \,{}_{\frac{1}{2}}\!\oplus \omega_i$, using Lemma 7.2(6), it must be that $o' = \frac{1}{2}o_i + \frac{1}{2}\vec{\omega_i}$ for some $o_i \in \mathit{Apply}^\Omega(T_i, \Delta_i)$ with $o_i \leq v_i$. By induction, $\Delta_i \models \phi_i$ for all $i \in I$, Therefore, by Definition 6.1, $\Delta \models \varphi$.

In case $\varphi \in \mathcal{L}$, $\varphi$ cannot be of the form $\mathbf{ref}(X)$. Then it is easy to show that $\sum_{\omega\in\Omega} v_\varphi(\omega) = 1$ and for all $\Delta$ and $o \in \mathit{Apply}^\Omega(T_\varphi, \Delta)$ we have $\sum_{w\in\Omega} o(\omega) = 1$. Therefore, $o \leq v_\varphi$ iff $o \geq v_\varphi$ iff $o = v_\varphi$, yielding (29). □

Completeness of $\sqsubseteq^\Omega_{pmay}$ and $\sqsubseteq^\Omega_{pmust}$, and hence also $\sqsubseteq_{pmay}$ and $\sqsubseteq_{pmust}$ by Theorem 6.7 and Theorem 3.3, follows from Lemma 7.4.

**Theorem 7.5.** 1. If $P \sqsubseteq^\Omega_{pmay} Q$ then $P \sqsubseteq_{\mathcal{L}} Q$.
2. If $P \sqsubseteq^\Omega_{pmust} Q$ then $P \sqsubseteq_{\mathcal{F}} Q$.

*Proof.* Suppose $P \sqsubseteq^\Omega_{pmay} Q$ and $[\![P]\!] \models \psi$ for some $\psi \in \mathcal{L}$. Let $N = fn(P, \psi)$ and let $T_{\langle N,\psi\rangle}$ be a characteristic test of $\psi$ with target value $v_\psi$. Then by Lemma 7.4, we have

$$\exists o \in \mathit{Apply}^\Omega(T_{\langle N,\psi\rangle}, [\![P]\!]) : o \geq v_\psi.$$

But since $P \sqsubseteq^\Omega_{pmay} Q$, this means $\exists o' \in \mathit{Apply}^\Omega(T_{\langle N,\psi\rangle}, [\![Q]\!]) : o \leq o'$, and thus $o' \geq v_\psi$. So again, by Lemma 7.4, we have $[\![Q]\!] \models \psi$. The case for must preorder is similar, using the Smyth preorder. □

**Theorem 7.6.** 1. If $P \sqsubseteq_{pmay} Q$ then $P \sqsubseteq_S Q$.
2. If $P \sqsubseteq_{pmust} Q$ then $P \sqsubseteq_{FS} Q$.

## 8. Related and future work

There have been a number of previous works on probabilistic extensions of the $\pi$-calculus by Palamidessi et. al. [14, 3, 19]. One distinction between our formulation with that of Palamidessi et. al. is the fact that we consider an interpretation of probabilistic summation as distribution over state-based processes, whereas in those works, a process like $s \,{}_p\!\oplus t$ is considered as a proper process, which can evolve into the distribution $p \cdot \delta[s] + (1-p) \cdot \delta[t]$ via an internal transition. We could encode this behaviour by a simple prefixing with the $\tau$ prefix. It would be interesting to see whether similar characterisations could be obtained for this restricted calculus. As far as we know, there are no existing works in the literature that give characterisations of the may- and must-testing preorders for the probabilistic $\pi$-calculus.

We structure our completeness proofs for the simulation preorders along the line of the proofs of similar characterisations of simulation preorders for pCSP [9, 7]. The name-passing feature of the $\pi$-calculus, however, gives rise to several complications not encountered in pCSP, and requires new techniques to deal with. In particular, due to the possibility of scope extrusion and close communication, the congruence properties of (failure) simulation is proved using an adaptation of the up-to techniques [22].

The immediate future work is to consider replication/recursion, for which we will need an advanced notion of weak transitions and consider divergence carefully, as in [8]. Moreover, we also have to limit ourselves to finite-state and finitely branching systems in order to characterise testing preorders by simulation preorders.

**Acknowledgment** Deng was partially supported by Natural Science Foundation of China (61173033, 61033002,61100053), and the Qatar National Research Fund under grant NPRP 09-1107-1-168 when he was visiting Carnegie Mellon University in 2011. Tiu is supported by the Australian Research Council Discovery Project DP110103173. Part of this work was done when he was visiting NICTA Kensington Lab in 2009; he would like to thank NICTA for the support he received during his visit.

## References

[1] L. Acciai, M. Boreale, and R. De Nicola. Linear-time and may-testing in a probabilistic reactive setting. In *FMOODS/FORTE*, volume 6722 of *Lecture Notes in Computer Science*, pages 29–43. Springer, 2011.

[2] M. Boreale and R. De Nicola. Testing equivalence for mobile processes. *Inf. Comput.*, 120(2):279–303, 1995.

[3] K. Chatzikokolakis and C. Palamidessi. A framework for analyzing probabilistic protocols and its application to the partial secrets exchange. *Theor. Comput. Sci.*, 389(3):512–527, 2007.

[4] L. Cheung, M. Stoelinga, and F. Vaandrager. A testing scenario for probabilistic processes. *J. ACM*, 54(6), Dec. 2007.

[5] R. De Nicola and M. Hennessy. Testing equivalences for processes. *Theor. Comput. Sci.*, 34:83–133, 1984.

[6] Y. Deng, R. van Glabbeek, C. Morgan, and C. Zhang. Scalar outcomes suffice for finitary probabilistic testing. In *ESOP*, volume 4421 of *LNCS*, pages 363–378. Springer, 2007.

[7] Y. Deng, R. J. van Glabbeek, M. Hennessy, and C. Morgan. Characterising testing preorders for finite probabilistic processes. *Logical Methods in Computer Science*, 4(4), 2008.

[8] Y. Deng, R. J. van Glabbeek, M. Hennessy, and C. Morgan. Testing finitary probabilistic processes. In *CONCUR*, volume 5710 of *LNCS*, pages 274–288. Springer, 2009.

[9] Y. Deng, R. J. van Glabbeek, M. Hennessy, C. Morgan, and C. Zhang. Remarks on testing probabilistic processes. *ENTCS*, 172:359–397, 2007.

[10] G. L. Ferrari, U. Montanari, and P. Quaglia. The weak late pi-calculus semantics as observation equivalence. In *CONCUR*, volume 962 of *Lecture Notes in Computer Science*, pages 57–71. Springer, 1995.

[11] H. Hansson and B. Jonsson. A calculus for communicating systems with time and probabilities. In *IEEE Real-Time Systems Symposium*, pages 278–287, 1990.

[12] M. Hennessy. Powerdomains and nondeterministic recursive definitions. In *Symposium on Programming*, volume 137 of *LNCS*, pages 178–193. Springer, 1982.

[13] M. Hennessy. *Algebraic Theory of Processes*. MIT Press, 1988.

[14] O. M. Herescu and C. Palamidessi. Probabilistic asynchronous pi-calculus. In *FoSSaCS*, volume 1784 of *LNCS*, pages 146–160. Springer, 2000.

[15] C. Hoare. *Communicating Sequential Processes*. Prentice-Hall, 1985.

[16] A. Ingólfsdóttir. Late and early semantics coincide for testing. *Theor. Comput. Sci.*, 146(1&2):341–349, 1995.

[17] R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes, II. *Inf. Comput.*, 100(1):41–77, 1992.

[18] R. Milner, J. Parrow, and D. Walker. Modal logics for mobile processes. *Theor. Comput. Sci.*, 114(1):149–171, 1993.

[19] G. Norman, C. Palamidessi, D. Parker, and P. Wu. Model checking probabilistic and stochastic extensions of the pi-calculus. *IEEE Trans. Software Eng.*, 35(2):209–223, 2009.

[20] A. Parma and R. Segala. Logical characterizations of bisimulations for dis- crete probabilistic systems. In *FOSSACS*, volume 4423 of *Lecture Notes in Computer Science*, pages 287–301. Springer, 2007.

[21] D. Sangiorgi. Bisimulation for higher-order process calculi. *Inf. Comput.*, 131(2):141–178, 1996.

[22] D. Sangiorgi. On the bisimulation proof method. *Mathematical Structures in Computer Science*, 8(5):447–479, 1998.

[23] D. Sangiorgi and D. Walker. *$\pi$-Calculus: A Theory of Mobile Processes*. Cambridge University Press, 2001.

[24] R. Segala. Testing probabilistic automata. In *CONCUR*, volume 1119 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 1996.

[25] R. Segala and N. A. Lynch. Probabilistic simulations for probabilistic processes. In *CONCUR*, volume 836 of *LNCS*, pages 481–496. Springer, 1994.

[26] R. J. van Glabbeek, S. A. Smolka, and B. Steffen. Reactive, generative and stratified models of probabilistic processes. *Inf. Comput.*, 121(1):59–80, 1995.

[27] R. J. van Glabbeek and W. P. Weijland. Branching time and abstraction in bisimulation semantics. *J. ACM*, 43(3):555–600, 1996.

[28] W. Yi and K. G. Larsen. Testing probabilistic and nondeterministic processes. In *PSTV*, volume C-8 of *IFIP Transactions*, pages 47–61. North-Holland, 1992.