Characterising Testing Preorders for Finite Probabilistic Processes

Yuxin Deng* Rob van Glabbeek Matthew Hennessy Carroll Morgan* Chenyi Zhang

Jiao Tong University, Shanghai, China
National ICT Australia, Sydney, Australia
School of CSE, University of New South Wales, Sydney, Australia
Department of Informatics, University of Sussex, Falmer, Brighton, UK
National ICT Australia
School of Computer Science and Engineering, The University of New South Wales, Sydney, Australia

Abstract

In 1992 Wang & Larsen extended the may- and must preorders of De Nicola and Hennessy to processes featuring probabilistic as well as nondeterministic choice. They concluded with two problems that have remained open throughout the years, namely to find complete axiomatisations and alternative characterisations for these preorders. This paper solves both problems for finite processes with silent moves. It characterises the may preorder in terms of simulation, and the must preorder in terms of failure simulation. It also gives a characterisation of both preorders using a modal logic. Finally it axiomatises both preorders over a probabilistic version of CSP.

1 Introduction

A satisfactory semantic theory for processes which encompass both nondeterministic and probabilistic behaviour has been a long-standing research problem [12, 39, 33, 23, 28, 1, 18, 24, 27, 37, 8]. For example, in 1992 Wang & Larsen posed the problems of finding complete axiomatisations and alternative characterisations for a natural generalisation of the standard testing preorders [7] to such processes. In this paper we solve both problems, at least for finite processes, by providing a detailed account of both may- and must testing preorders for a finite version of the process calculus CSP extended with probabilistic choice. For each preorder we provide three independent characterisations, using (i) coinductive simulation relations, (ii) a modal logic and (iii) sets of inequations.

Testing processes: Our starting point is the finite process calculus pCSP [9] obtained by adding a probabilistic choice operator to finite CSP; like others who have done the same, we now have *three* choice operators, external $P \,\Box\, Q$, internal $P \sqcap Q$ and the newly added probablistic choice $P \,{}_p\!\oplus\, Q$. So a semantic theory for pCSP will have to provide a coherent account of the precise relationships between these operators.

As a first step, in Sec. 2 we provide an interpretation of pCSP as a *probabilistic labelled transition system* [35], in which transitions like $s \xrightarrow{\alpha} s'$ from standard labelled transition systems are generalised to the form $s \xrightarrow{\alpha} \Delta$, where Δ is a *distribution*, a mapping which assigns probabilities to states. With this interpretation we can now obtain a version of the testing preoders [7] for pCSP processes, $P \sqsubseteq_{\text{pmay}} Q$ and $P \sqsubseteq_{\text{pmust}} Q$; see Sec. 3. These are based on the ability of processes to pass *tests*; the tests we use are simply pCSP processes in which certain *states* have been marked as *success states*. See [9] for a detailed discussion of the power of such tests.

The object of this paper is to give useful characterisations of these testing preorders. Our first problem is that in the literature there is considerable variation in the manner in which the standard notion of testing [7] is generalised to probabilistic processes. For example in [10], following [34], the success of a test is indicated by the *actual execution* of a single predefined *success action*. Luckily, as we show in Sec. 5, both approaches, our *state* based approach and this *action* based approach, coincide for pCSP [1]. As another variation [34] allows the use of a countable number of *success actions*, so-called *vector-based testing*; indeed Segala's results in [34] depend crucially on this form of test. However, in the recent work [10], it has been shown that at least for finite processes the resulting preorders again coincide with $\sqsubseteq_{\text{pmay}}$ and $\sqsubseteq_{\text{pmust}}$; this is reported in Sec. 6 below. Later in the paper, following in Segala's footsteps, we make extensive use of this technically more convenient vector-based testing.

*We acknowledge the support of the Australian Research Council (ARC) Grant DP034557.

[1] However in the presence of divergence they lead to slightly different preorders.

Simulation preorders: In Sec. 4 we use the transitions $s \xrightarrow{\alpha} \Delta$ to define two co-inductive preorders, the *simulation* preorder $P \sqsubseteq_S Q$ [24, 9], and the novel *failure simulation* preorder $P \sqsubseteq_{FS} Q$ over pCSP processes. Their definition uses a natural generalisation of the transitions, first (Kleisli-style) to take the form $\Delta \xrightarrow{\alpha} \Delta'$, and then to *weak* versions $\Delta \stackrel{\alpha}{\Longrightarrow} \Delta$. The latter failure simulation preorder differs from the former in the use of a *failure* predicate $s \stackrel{X}{\not\longrightarrow}$, indicating that in the state s none of the actions in X can be performed.

It can be shown that both preorders are preserved by all the operators in pCSP, and that they are *sound* with respect to the testing preorders; that is $P \sqsubseteq_S Q$ implies $P \sqsubseteq_{\text{pmay}} Q$ and $P \sqsubseteq_{FS} Q$ implies $P \sqsubseteq_{\text{pmust}} Q$. The proofs are long, and rely on a series of properties of the derived transitions $\Delta \stackrel{\alpha}{\Longrightarrow} \Delta'$. Nevertheless, they follow a well-worn pattern and are omitted from this extended abstract. But *completeness*, that the testing preorders imply the respective simulation preorders, requires some ingenuity. We prove it indirectly, by using a characterisation of the simulation preorders in terms of a modal logic.

Modal logic: Our modal logic, defined in Sec. 7, uses finite conjunction $\bigwedge_{i\in I} \varphi_i$, a *next state* construct $\langle a\rangle\varphi$, and a novel probabilistic construct $\bigoplus_{i\in I} p_i \cdot \varphi_i$; in addition, to capture failures, we have, for every set of actions X, a corresponding formula $\mathbf{ref}(X)$, to be satisfied by any process which can not perform any of the actions in X. A satisfaction relation between processes and formula then gives, in a natural manner, a *logical preorder* between processes: $P \sqsubseteq^{\mathcal{L}} Q$ means that every $\mathcal{L}$-formula satisfied by Q is also satisfied by P.

This leads to our first characterisation result, namely that the failure simulation preorder is determined by our modal logic: $P \sqsubseteq_{\text{pmust}} Q$ if and only if $P \sqsubseteq^{\mathcal{L}} Q$; the proof involves constructing, for each pCSP process P, a *characteristic formula* φ_P. A similar characterisation for $P \sqsubseteq_{\text{pmay}} Q$ can be obtained by dropping the formulae $\mathbf{ref}(X)$ from the modal logic.

The modal logic also supplies the key to *completeness*, that the testing preorders imply the simulations preorders, leading to our second set of characterisation results. In Sec. 8 we show how every modal formula φ can be captured, in some sense, by a test T_φ; essentially the ability of a pCSP process to satisfy φ is determined by its ability to pass the test T_φ. Incidently it is useful here to be able to use *vector-based tests*.

Equations: It is well-known that may- and must testing for standard CSP can be captured equationally [7, 3, 14]. In [9] we showed that most of the standard equations are no longer valid in the probabilistic setting of pCSP; we also gave a *partial* result, providing a set of axioms which are complete with respect to (probabilistic) may-testing for the sub-language of pCSP which contains no probabilistic choices. Here we extend this result, by showing, in Sec. 10, that both $P \sqsubseteq_{\text{pmay}} Q$ and $P \sqsubseteq_{\text{pmust}} Q$ can still be captured equationally over full pCSP. In the may case the essential (in)equation required is

$$a.(P \,{}_p\!\oplus\, Q) \quad \sqsubseteq \quad a.P \,{}_p\!\oplus\, a.Q$$

The must case is more complicated: in the absence of the distributivity of the external and internal choices over each other, to obtain completeness we require an inequational schema, refered to as $(\mathbf{Must2})$ in Fig. 3 below.

This completes our introduction to the paper; related work is discussed in the final section, Sec. 12.

2 Finite probabilistic CSP

Let Act be a finite set of actions, ranged over by $a, b, \cdots$, which processes can perform. Then the finite probabilistic CSP processes are given by the following two-sorted syntax:

$$\begin{array}{lcl} P & ::= & S \mid P \,{}_p\!\oplus\, P \\ S & ::= & \mathbf{0} \mid a.P \mid P \sqcap P \mid S \,\Box\, S \mid S \,|_A\, S \end{array}$$

Here $P \,{}_p\!\oplus\, Q$, for $0 \le p \le 1$, represents a *probabilistic choice* between P and Q: with probability p it will act like P and with probability $1-p$ it will act like Q. Any process is a probabilistic combination of state-based processes (the sub-sort S above) built by repeated application of the operator ${}_p\!\oplus$. The state-based processes have a CSP-like syntax, involving the stopped process $\mathbf{0}$, action prefixing $a._-$, *internal-* and *external choices* $\sqcap$ and $\Box$, and a *parallel composition* $|_A$ for $A \subseteq$ Act.

The process $P \sqcap Q$ will first do a so-called *internal action* $\tau \notin$ Act, choosing *nondeterministically* between P and Q. Therefore $\sqcap$, like $a._-$, acts as a *guard*, in the sense that it converts any process arguments into a state-based process.

The process $P \,\Box\, Q$ on the other hand does not perform actions itself, but merely allows its arguments to proceed, disabling one argument as soon as the other has done a visible action. In order for this process to start from a state rather than a probability distribution of states, we require its arguments to be state-based as well; the same applies to $|_A$. Expressions $P \,\Box\, Q$ and $P \,|_A\, Q$ for processes P and Q that are *not* state-based are therefore syntactic sugar for an expression in the above syntax obtained by distributing $\Box$ and $|_A$ over ${}_p\!\oplus$.

Finally, the expression $P \,|_A\, Q$, where $A \subseteq$ Act, represents processes P and Q running in parallel. They may synchronise by performing the same action from A simultaneously; such a synchronisation results in τ. In addition P and Q may independently do any action from $(\text{Act}\backslash A) \cup \{\tau\}$.

We write pCSP for the set of process terms defined by this grammar, and sCSP for the subset comprising only the state-based process terms. The full language of CSP [3, 15, 31] has many more operators; we have simply chosen a representative selection, and have added probabilistic choice. Our parallel operator is not a CSP primitive, but it can easily be expressed in terms of them—in particular $P \mid_A Q = (P \| _A Q) \backslash A$, where $\|_A$ and $\backslash A$ are the parallel composition and hiding operators of [31]. It can also be expressed in terms of the parallel composition, renaming and restriction operators of CCS. We have chosen this (non-associative) operator for convenience in defining the application of tests to processes.

As usual we may elide $\mathbf{0}$; the prefixing operator $a._-$ binds stronger than any binary operator; and precedence between binary operators is indicated via brackets or spacing. We will also sometimes use indexed binary operators, such as $\bigoplus_{i\in I} p_i{\cdot}P_i$ with $\sum_{i\in I} p_i = 1$, and $\prod_{i\in I} P_i$.

The above intuitions are formalised by an *operational semantics* associating with each process term a graph-like structure representing its possible reactions to users' requests: we use a generalisation of labelled transition systems [25] that includes probabilities.

A (discrete) probability distribution over a set S is a function $\Delta\colon S \to [0,1]$ with $\sum_{s\in S}\Delta(s) = 1$; the *support* of Δ is given by $\lceil\Delta\rceil = \{\, s \in S \mid \Delta(s) > 0 \,\}$. We write $\mathcal{D}(S)$, ranged over by Δ, Θ, Φ, for the set of all distributions over S with finite support; these finite distributions are sufficient for the results of this paper. We also write $\overline{s}$ to denote the point distribution assigning probability 1 to s and 0 to all others, so that $\lceil\overline{s}\rceil = \{s\}$.

For Δ a distribution over S and function $f\colon S \to X$ into a linear vector space X (typically the reals[2]) we write $\bigoplus_{s\in S} \Delta_s{\cdot}f_s$ or $\mathrm{Exp}_\Delta(f)$ for $\sum_{s\in S}\Delta(s){\cdot}f(s)$, the *weighted average* of the f_s, or *expected value* of f. When $p\in[0,1]$, we also write $f_1 \,{}_p\!\oplus\, f_2$ for $p\cdot f_1 + (1{-}p)\cdot f_2$. More generally, for function $F : S \to \mathcal{P}^+(X)$ with $\mathcal{P}^+(X)$ being the collection of non-empty subsets of X, we define $\mathrm{Exp}_\Delta F := \{\, \mathrm{Exp}_\Delta(f) \mid f \mathrel{\overline{\in}} F \,\}$, where $f \mathrel{\overline{\in}} F$ means that f is a *choice function* with $f(s) \in F(s)$ for all $s \in S$.

We now give the probabilistic generalisation (pLTSs) of labelled transition systems (LTSs):

Definition 1 A *probabilistic labelled transition system* is a triple $\langle S, \mathsf{Act}_\tau, \to\rangle$, where

(i) S is a set of states

(ii) Act_τ is a set of actions Act, augmented by $\tau \notin \mathsf{Act}$; we let a range over Act and α over Act_τ.

(iii) relation $\to$ is a subset of $S \times \mathsf{Act}_\tau \times \mathcal{D}(S)$.

As with LTSs, we usually write $s \xrightarrow{\alpha} \Delta$ for $(s,\alpha,\Delta) \in \to$, $s \xrightarrow{\alpha}$ for $\exists\Delta : s \xrightarrow{\alpha} \Delta$ and $s \to$ for $\exists\alpha : s \xrightarrow{\alpha}$. An LTS may be viewed as a degenerate pLTS, one in which only point distributions are used.

[2] Other possibilities are tuples of reals, or distributions over some set.

$$a.P \xrightarrow{a} [\![P]\!]$$

$$P \sqcap Q \xrightarrow{\tau} [\![P]\!] \qquad P \sqcap Q \xrightarrow{\tau} [\![Q]\!]$$

$$\frac{s_1 \xrightarrow{a} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{a} \Delta} \qquad \frac{s_2 \xrightarrow{a} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{a} \Delta}$$

$$\frac{s_1 \xrightarrow{\tau} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{\tau} \Delta \,\Box\, s_2} \qquad \frac{s_2 \xrightarrow{\tau} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{\tau} s_1 \,\Box\, \Delta}$$

$$\frac{s_1 \xrightarrow{\alpha} \Delta \qquad \alpha \notin A}{s_1 \mid_A s_2 \xrightarrow{\alpha} \Delta \mid_A s_2} \qquad \frac{s_2 \xrightarrow{\alpha} \Delta \qquad \alpha \notin A}{s_1 \mid_A s_2 \xrightarrow{\alpha} s_1 \mid_A \Delta}$$

$$\frac{s_1 \xrightarrow{a} \Delta_1,\ s_2 \xrightarrow{a} \Delta_2 \quad a \in A}{s_1 \,\Box\, s_2 \xrightarrow{\tau} \Delta_1 \mid_A \Delta_2}$$

Figure 1. Operational semantics of pCSP.

We now define the operational semantics of pCSP by means of a particular pLTS $\langle \mathsf{sCSP}, \mathsf{Act}_\tau, \to\rangle$, constructed by taking sCSP to be the set of states and interpreting pCSP processes P as distributions $[\![P]\!] \in \mathcal{D}(\mathsf{sCSP})$ as follows:

$$\begin{array}{rcl} [\![s]\!] & := & \overline{s} \quad \text{for } s \in \mathsf{sCSP} \\ [\![P \,{}_p\!\oplus\, Q]\!] & := & [\![P]\!] \,{}_p\!\oplus\, [\![Q]\!] \,. \end{array}$$

Note that for each $P \in \mathsf{pCSP}$ the distribution $[\![P]\!]$ is finite, i.e. it has finite support. The definition of the relations $\xrightarrow{\alpha}$ is given in Fig. 1. These rules are very similar to the standard ones used to interpret CSP as an LTS [31], but modified so that the result of an action is a distribution.

3 Testing pCSP processes

A *test* is a pCSP process except that it may have subterms $\omega.P$ for fresh $\omega \notin \mathsf{Act}_\tau$, a special action reporting success; and the operational semantics above is extended by treating ω like any other action from Act. To apply test T to process P we form the process $T \mid_{\mathsf{Act}} P$ in which *all* visible actions of P must synchronise with T, and define a set of testing outcomes $\mathcal{A}(T,P)$ where each outcome, in $[0,1]$, arises from a resolution of the nondeterministic choices in $T \mid_{\mathsf{Act}} P$ and gives the probability that this resolution will reach a *success state*, one in which ω is possible.

To this end, we inductively define a *results-gathering* function $\mathbb{V} : S \to \mathcal{P}^+([0,1])$; it extends to type $\mathcal{D}(S) \to \mathcal{P}^+([0,1])$ via the convention $\mathbb{V}(\Delta) := \mathrm{Exp}_\Delta \mathbb{V}$.

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

Note that these choices are exhaustive because $T \mid_{\mathsf{Act}} P$ has only τ, ω actions, and that $\mathbb{V}$ is well defined when applied to finite, loop-free pLTSs, such those of pCSP.

Definition 2 For any pCSP process P and test T, define

$$\mathcal{A}(T,P) \;:=\; \mathbb{V}[\![T \mid_{\mathsf{Act}} P]\!] \,.$$

With this definition, the general testing framework of [7] yields two testing preorders for pCSP, one based on *may* testing, written $P \sqsubseteq_{\text{pmay}} Q$, and the other on *must* testing, written $P \sqsubseteq_{\text{pmust}} Q$.

Definition 3 The *may*- and *must* preorders are given by

$$\begin{array}{rcl} P \sqsubseteq_{\text{pmay}} Q & \text{iff} & \forall \text{ tests } T\colon \mathcal{A}(T,P) \leq_{\text{Ho}} \mathcal{A}(T,Q) \\ P \sqsubseteq_{\text{pmust}} Q & \text{iff} & \forall \text{ tests } T\colon \mathcal{A}(T,P) \leq_{\text{Sm}} \mathcal{A}(T,Q) \end{array}$$

with $\leq_{\text{Ho}}, \leq_{\text{Sm}}$ the Hoare, Smyth preorders on $\mathcal{P}^+[0,1]$.[3]

4 Simulation and failure simulation

Let $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ be a relation from states to distributions. We lift it to a relation $\overline{\mathcal{R}} \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$ by letting $\Delta \,\overline{\mathcal{R}}\, \Theta$ whenever there is an index set I and $p \in \mathcal{D}(I)$ such that

(i) $\Delta = \bigoplus_{i \in I} p_i \cdot \overline{s_i}$,

(ii) For each $i \in I$ there is a distribution Φ_i s.t. $s_i \;\mathcal{R}\; \Phi_i$,

(iii) $\Theta = \bigoplus_{i \in I} p_i \cdot \Phi_i$.

For notational convenience, the lifted version of the transition relations $\xrightarrow{\alpha}$ for $\alpha \in \mathsf{Act}_\tau$ are again denoted $\xrightarrow{\alpha}$.

We write $s \xrightarrow{\hat{\tau}} \Delta$ if either $s \xrightarrow{\tau} \Delta$ or $\Delta = \overline{s}$; again $\Delta_1 \xrightarrow{\hat{\tau}} \Delta_2$ denotes the lifted relation. Thus e.g. we have $[\![(a \sqcap b) \,{}_{\frac{1}{2}}\!\oplus\, (a \sqcap c)]\!] \xrightarrow{\hat{\tau}} [\![a \,{}_{\frac{1}{2}}\!\oplus\, ((a \sqcap b) \,{}_{\frac{1}{2}}\!\oplus\, c)]\!]$ [9].

We now define the weak transition relation $\stackrel{\hat{\tau}}{\Longrightarrow}$ as the transitive and reflexive closure $\xrightarrow{\hat{\tau}}{}^*$ of $\xrightarrow{\hat{\tau}}$, while for $a \neq \tau$ $\Delta_1 \stackrel{\hat{a}}{\Longrightarrow} \Delta_2$ denotes $\Delta_1 \stackrel{\hat{\tau}}{\Longrightarrow} \xrightarrow{a} \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_2$. We write $s \not\xrightarrow{X}$ with $X \subseteq \mathsf{Act}$ when $\forall \alpha \in X \cup \{\tau\} : s \not\xrightarrow{\alpha}$, and $\Delta \not\xrightarrow{X}$ when $\forall s \in \lceil \Delta \rceil : s \not\xrightarrow{X}$.

Definition 4 A relation $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ is said to be a *failure simulation* if for all $s, \Theta, \alpha, \Delta$ we have that

- $s \;\mathcal{R}\; \Theta \wedge s \xrightarrow{\alpha} \Delta$ implies $\exists \Theta' \colon \Theta \stackrel{\hat{\alpha}}{\Longrightarrow} \Theta' \wedge \Delta \;\overline{\mathcal{R}}\; \Theta'$
- $s \;\mathcal{R}\; \Theta \wedge s \not\xrightarrow{X} \Delta$ implies $\exists \Theta' \colon \Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \wedge \Theta' \not\xrightarrow{X}$.

We write $s \rhd_{\mathit{FS}} \Theta$ to mean that there is some failure simulation $\mathcal{R}$ such that $s \;\mathcal{R}\; \Theta$. Similarly, we define *simulation* and $s \rhd_S \Theta$ by dropping the second clause in Def. 4.

[3] The Hoare order is defined by $X \leq_{\text{Ho}} Y$ iff $\forall x \in X\colon \exists y \in Y\colon x \leq y$, similarly the Smyth order by $X \leq_{\text{Sm}} Y$ iff $\forall y \in Y\colon \exists x \in X\colon x \leq y$.

Definition 5 The *simulation preorder* $\sqsubseteq_S$ and *failure simulation preorder* $\sqsubseteq_{FS}$ on pCSP are defined as follows:

$$\begin{array}{rcl} P \sqsubseteq_S Q & \text{iff} & [\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \text{ for some } \Theta \text{ with } [\![P]\!] \;\overline{\rhd_S}\; \Theta \\ P \sqsubseteq_{FS} Q & \text{iff} & [\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Theta \text{ for some } \Theta \text{ with } [\![Q]\!] \;\overline{\rhd_{\mathit{FS}}}\; \Theta \,. \end{array}$$

(Note the opposing directions.) The symmetric closures of $\sqsubseteq_S$ and $\sqsubseteq_{FS}$ are called *(failure) simulation equivalence*, denoted $\simeq_S$ and $\simeq_{FS}$, respectively.

We have already shown in [9] that $\sqsubseteq_S$ is a precongruence and that it implies $\sqsubseteq_{\text{pmay}}$. Similar results can be established for $\sqsubseteq_{FS}$ as well. We summarise these facts as follows:

Proposition 1 Suppose $\sqsubseteq \in \{\sqsubseteq_S, \sqsubseteq_{FS}\}$. Then $\sqsubseteq$ is a preorder, and if $P_i \sqsubseteq Q_i$ for $i = 1, 2$ then $a.P_1 \sqsubseteq a.Q_1$ for $a \in \mathsf{Act}$ and $P_1 \odot P_2 \;\sqsubseteq\; Q_1 \odot Q_2$ for $\odot \in \{\sqcap, \;\Box, {}_p\!\oplus, \mid_A\}$.

Proof: The case $\sqsubseteq_S$ was proved in [9, Cor. 6.10 and Thm. 6.13]; the case $\sqsubseteq_{FS}$ is analogous. □

Theorem 1

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

Proof: The first clause was proved in [9, Thm. 6.17]; the second can be shown similarly. □

The next four sections are devoted to obtaining the converse.

5 State- versus action-based testing

Much work on testing [7, 39, 9] uses success *states* marked by outgoing ω-actions; in other work [34, 10], however, it is the *actual execution* of ω that constitutes success. The former, *state-based* testing, leads to the preorders we defined in Sec. 3; the latter, *action-based* testing, leads to slightly different preorders $\widehat{\sqsubseteq}_{\text{may}}$ and $\widehat{\sqsubseteq}_{\text{must}}$. Without probability there is no difference between $\widehat{\sqsubseteq}_{\text{may}}$ and $\sqsubseteq_{\text{may}}$; but possible divergence makes $\sqsubseteq_{\text{must}}$ strictly more discriminating than $\widehat{\sqsubseteq}_{\text{must}}$, and in fact $\sqsubseteq_{\text{must}}$ coincides with CSP refinement based on failures and divergences [3, 15, 31]. The action-based approach is formalised as in the state-based approach, via a suitable $\widehat{\mathbb{V}}$:

$$\widehat{\mathbb{V}}(s) := \begin{cases} \bigcup\{\widehat{\mathbb{V}}(\Delta) \mid s \xrightarrow{\tau} \Delta\} \cup \{1 \mid s \xrightarrow{\omega}\} & \text{if } s \rightarrow \\ \{0\} & \text{otherwise} \end{cases}$$

Proposition 2

1. If $P \sqsubseteq_{\text{pmay}} Q$ then $P \;\widehat{\sqsubseteq}_{\text{pmay}}\; Q$.
2. If $P \sqsubseteq_{\text{pmust}} Q$ then $P \;\widehat{\sqsubseteq}_{\text{pmust}}\; Q$.

Proof: For any test $\widehat{T}$ construct T by replacing each subterm $\omega.Q$ by $\tau.\omega$; then $\mathbb{V}[\![T \mid_{\mathsf{Act}} P]\!] = \widehat{\mathbb{V}}[\![\widehat{T} \mid_{\mathsf{Act}} P]\!]$ for all pCSP processes P. □

In fact we use the action-based preorders in Thm. 2 below, a (quasi-, thus) converse of Thm. 1; but with Prop. 2 above the two kinds of preorders become identified so that Thms. 1 and 2 are converse to each other.

Theorem 2

1. If $P \,\widehat{\sqsubseteq}_{\text{pmay}}\, Q$ then $P \sqsubseteq_S Q$.
2. If $P \,\widehat{\sqsubseteq}_{\text{pmust}}\, Q$ then $P \sqsubseteq_{FS} Q$. □

We set this theorem as our goal in the next three sections.

6 Vector-based testing

This section describes another variation on testing, a richer testing framework due to Segala [34], in which countably many success actions exist: the application of a test to a process yields a set of *vectors* over the real numbers, rather than a set of scalars. The resulting action-based testing preorders will serve as a stepping stone in proving Thm. 2.

Let Ω be a *set* of fresh success actions with $\Omega \cap \mathsf{Act}_\tau = \emptyset$. An Ω-test is again a pCSP process, but this time allowing subterms $\omega.P$ for any $\omega \in \Omega$. Applying such a test to a process yields a non-empty set of test outcome-*tuples* $\widehat{\mathcal{A}}^\Omega(T,P) \subseteq [0,1]^\Omega$. Each *tuple* arises from a resolution within $T \mid_{\mathsf{Act}} P$ of nondeterministic choices into probabilistic choices, and its ω-component gives the probability that this resolution will perform the success action ω.

For vectors we again inductively define a results-gathering function $\widehat{\mathbb{V}}^\Omega_\updownarrow : S \to \mathcal{P}^+[0,1]^\Omega$; it extends to type $\mathcal{D}(S) \to \mathcal{P}^+[0,1]^\Omega$ via $\widehat{\mathbb{V}}^\Omega_\updownarrow(\Delta) := \text{Exp}_\Delta \widehat{\mathbb{V}}^\Omega_\updownarrow$ just as $\mathbb{V}$ and $\widehat{\mathbb{V}}$ did. First, for any α define $\alpha! : [0,1]^\Omega \to [0,1]^\Omega$ so that $\alpha!o(\omega)$ becomes 1 if $\omega{=}\alpha$ but remains $o(\omega)$ otherwise; this function lifts to sets $O \subseteq [0,1]^\Omega$ as usual, via $\alpha!O := \{\alpha!o \mid o \in O\}$. Now we define

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

where $\vec{0} \in [0,1]^\Omega$ is given by $\vec{0}(\omega) = 0$ for all $\omega \in \Omega$, and the *convex closure* $\updownarrow X$ of a set X is defined

$$\updownarrow X := \{\, \textstyle\bigoplus_{i\in I} p_i \cdot o_i \mid p \in \mathcal{D}(I) \text{ and } o : I \to X \,\} \quad .$$

We extend our earlier results-gathering definitions so that $\mathbb{V}_\updownarrow(s) := \updownarrow\mathbb{V}(s)$ and $\widehat{\mathbb{V}}_\updownarrow(s) := \updownarrow\widehat{\mathbb{V}}(s)$. Note that in case $\Omega := \{\omega\}$ we have $\widehat{\mathbb{V}}^\Omega_\updownarrow = \widehat{\mathbb{V}}_\updownarrow$, and the convex-closing preorders based on $\mathbb{V}_\updownarrow, \widehat{\mathbb{V}}_\updownarrow$ coincide with the simpler ones based on $\mathbb{V}, \widehat{\mathbb{V}}$. Thus convex closure matters only for proper vectors, as explained in the remark following Def. 6.

In [10] the results-gathering function $\widehat{\mathbb{V}}^\Omega_\updownarrow$ with $\Omega = \{\omega_1, \omega_2, \cdots\}$ was called simply $\mathbb{W}$ (because action-based/convex/vector-based testing was assumed there throughout, making the $\widehat{\cdot}^\Omega_\updownarrow$-indicators superfluous); and it was defined in terms of a formalisation of the notion of a resolution. The inductive definition above yields the same results.

Definition 6 For any pCSP process P and Ω-test T, let

$$\widehat{\mathcal{A}}^\Omega_\updownarrow(T,P) \;:=\; \widehat{\mathbb{V}}^\Omega_\updownarrow[\![T \mid_{\mathsf{Act}} P]\!] \,.$$

The *vector-based may-* and *must* preorders are given by

$$P \,\widehat{\sqsubseteq}^\Omega_{\text{pmay}}\, Q \;\text{ iff }\; \forall\, \Omega\text{-tests } T\text{: } \widehat{\mathcal{A}}^\Omega_\updownarrow(T,P) \leq_{\text{Ho}} \widehat{\mathcal{A}}^\Omega_\updownarrow(T,Q)$$
$$P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q \;\text{ iff }\; \forall\, \Omega\text{-tests } T\text{: } \widehat{\mathcal{A}}^\Omega_\updownarrow(T,P) \leq_{\text{Sm}} \widehat{\mathcal{A}}^\Omega_\updownarrow(T,Q)$$

where $\leq_{\text{Ho}}, \leq_{\text{Sm}}$ are the Hoare-, Smyth preorders on $\mathcal{P}^+[0,1]^\Omega$ generated from $\leq$ index-wise on $[0,1]^\Omega$ itself.

Remark: For proper vector-based testing, convex closure matters, as it allows internal choice to simulate probabilistic choice [13]. Consider the following two processes

$$P := a \sqcap b \sqcap (a \,{}_{\frac{1}{2}}\!\oplus\, b) \quad \text{ and } \quad Q := a \sqcap b.$$

It is obvious that $P \sqsubseteq_S Q$, and from Thm. 1 it therefore follows that $P \sqsubseteq_{\text{pmay}} Q$. However if $\Omega = \{\omega_1, \omega_2\}$ and we remove the convex closure in the definition of $\widehat{\mathbb{V}}^\Omega_\updownarrow$, then with the test $T := a.w_1 \,\Box\, b.w_2$ we would have

$$\widehat{\mathcal{A}}^\Omega(T,P) = \{(1,0),(0,1),(0.5,0.5)\}$$
$$\widehat{\mathcal{A}}^\Omega(T,Q) = \{(1,0),(0,1)\}$$

and so $\widehat{\mathcal{A}}^\Omega(T,P) \not\leq_{\text{Ho}} \widehat{\mathcal{A}}^\Omega(T,Q)$. However, their convex closures $\widehat{\mathcal{A}}^\Omega_\updownarrow(T,P)$ and $\widehat{\mathcal{A}}^\Omega_\updownarrow(T,Q)$ *are* related under the Hoare preorder. □

The testing preorders of [34] are obtained by taking Ω to be a countably infinite set, whereas the preorders $\widehat{\sqsubseteq}_{\text{pmay}}$ and $\widehat{\sqsubseteq}_{\text{pmust}}$ of Sec. 5 were obtained by taking Ω to be the singleton set $\{\omega\}$. In [10] we established that for our finite pCSP processes the two coincide:

Theorem 3 [10].

1. $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmay}}\, Q$ iff $P \,\widehat{\sqsubseteq}_{\text{pmay}}\, Q$
2. $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q$ iff $P \,\widehat{\sqsubseteq}_{\text{pmust}}\, Q$. □

Thus, with the *if*-direction of Thm. 3, for Thm. 2 it will suffice to show that $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmay}}\, Q$ implies $P \sqsubseteq_S Q$ and $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q$ implies $P \sqsubseteq_{FS} Q$. The crucial characteristics of $\widehat{\mathcal{A}}^\Omega_\updownarrow$ needed for that implication are summarised in this lemma (proof omitted):

Lemma 1 Let P be a pCSP process, and T, T_i be tests.

1. $o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(\omega, P)$ iff $o = \vec{\omega}$.

2. Suppose the action ω does not occur in the test T. Then $o \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(\omega \,\Box\, a.T, P)$ with $o(\omega) = 0$ iff there is a $\Delta \in \mathcal{D}(\mathsf{sCSP})$ with $[\![P]\!] \stackrel{\hat{a}}{\Longrightarrow} \Delta$ and $o \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T, \Delta)$.

3. $\vec{0} \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(\Box_{a\in X}\, a.\omega, P)$ iff $\exists \Delta : [\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta \stackrel{X}{\not\rightarrow}$.

4. $o \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(\bigoplus_{i\in I} p_i{\cdot}T_i, P)$ iff $o = \sum_{i\in I} p_i o_i$ for certain $o_i \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T_i, P)$.

5. $o \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(\prod_{i\in I} T_i, P)$ iff for all $i \in I$ there are $p_i \in [0,1]$ and $\Delta_i \in \mathcal{D}(\mathsf{sCSP})$ such that $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \bigoplus_{i\in I} p_i{\cdot}\Delta_i$ and $o = \bigoplus_{i\in I} p_i o_i$ for certain $o_i \in \widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T_i, \Delta_i)$.

Here $\vec{\omega} \in [0,1]^{\Omega}$ is given by $\vec{\omega}(\omega) = 1$ and $\vec{\omega}(\omega') = 0$ for $\omega' \neq \omega$. □

In writing $\widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T, \Delta)$ above we treat a distribution Δ as the pCSP expression $\bigoplus_{s\in\mathsf{sCSP}} \Delta_s{\cdot}s$; and as usual we define $\widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T, \Delta) := \mathrm{Exp}_{\Delta}\widehat{\mathcal{A}}^{\Omega}_{\updownarrow}(T, \_)$.

7 Modal logic

Our next step towards Thm. 2 is to define a set $\mathcal{L}$ of modal formulae, inductively, as follows:

- $\langle a\rangle\varphi \in \mathcal{L}$ when $\varphi \in \mathcal{L}$ and $a \in \mathsf{Act}$,
- $\mathbf{ref}(X) \in \mathcal{L}$ when $X \subseteq \mathsf{Act}$,
- $\bigwedge_{i\in I} \varphi_i \in \mathcal{L}$ when $\varphi_i \in \mathcal{L}$ for all $i \in I$, with I finite
- and $\bigoplus_{i\in I} p_i{\cdot}\varphi_i \in \mathcal{L}$ when $p_i \in [0,1]$ and $\varphi_i \in \mathcal{L}$ for all $i \in I$, with I a finite index set, and $\sum_{i\in I} p_i = 1$.

We often write $\varphi_1 \,{}_p\!\oplus\, \varphi_2$ for $\bigoplus_{i\in\{1,2\}} p_i{\cdot}\varphi_i$ with $p = p_1$, and $\varphi_1 \wedge \varphi_2$ for $\bigwedge_{i\in\{1,2\}} \varphi_i$ and finally $\top$ for $\bigwedge_{i\in\emptyset} \varphi_i$.

Let $\mathcal{L}^-$ be the subclass of $\mathcal{L}$ obtained by skipping the $\mathbf{ref}(X)$ clause. The *satisfaction relation* $\models \subseteq \mathcal{D}(\mathsf{sCSP}) \times \mathcal{L}$ is given by:

- $\Delta \models \langle a\rangle\varphi$ iff there is a Δ' with $\Delta \stackrel{\hat{a}}{\Longrightarrow} \Delta'$ and $\Delta' \models \varphi$,
- $\Delta \models \mathbf{ref}(X)$ iff there is a Δ' with $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \stackrel{X}{\not\rightarrow}$,
- $\Delta \models \bigwedge_{i\in I} \varphi_i$ iff $\Delta \models \varphi_i$ for all $i \in I$
- and $\Delta \models \bigoplus_{i\in I} p_i{\cdot}\varphi_i$ iff there are $\Delta_i \in \mathcal{D}(\mathsf{sCSP})$, for all $i \in I$, with $\Delta_i \models \varphi_i$, such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \bigoplus_{i\in I} p_i{\cdot}\Delta_i$.

We write $\Delta \sqsubseteq^{\mathcal{L}^-} \Theta$ just when $\Delta \models \varphi$ implies $\Theta \models \varphi$ for all $\varphi \in \mathcal{L}^-$; we write $\Delta \sqsubseteq^{\mathcal{L}} \Theta$ just when $\Delta \models \varphi$ is implied by $\Theta \models \varphi$ for all $\varphi \in \mathcal{L}$. (Note the opposing directions.)

Definition 7 The *characteristic formula* φ_s or φ_Δ of a process $s \in \mathsf{sCSP}$ or $\Delta \in \mathcal{D}(\mathsf{sCSP})$ is defined inductively:

- $\varphi_s := \bigwedge_{s\stackrel{a}{\longrightarrow}\Delta} \langle a\rangle\varphi_\Delta \wedge \mathbf{ref}(\{a \mid s \stackrel{a}{\not\rightarrow}\})$ if $s \stackrel{\tau}{\not\rightarrow}$,
- $\varphi_s := \bigwedge_{s\stackrel{a}{\longrightarrow}\Delta} \langle a\rangle\varphi_\Delta \wedge \bigwedge_{s\stackrel{\tau}{\longrightarrow}\Delta} \varphi_\Delta$ otherwise,
- $\varphi_\Delta := \bigoplus_{s\in\lceil\Delta\rceil} \Delta(s){\cdot}\varphi_s$.

Here the conjunctions $\bigwedge_{s\stackrel{a}{\longrightarrow}\Delta}$ range over suitable pairs a, Δ, and $\bigwedge_{s\stackrel{\tau}{\longrightarrow}\Delta}$ ranges over suitable Δ.

Write $\varphi \Rightarrow \psi$ with $\varphi, \psi \in \mathcal{L}$ if for each distribution Δ one has $\Delta \models \varphi$ implies $\Delta \models \psi$. Then it is easy to see that $\varphi_{\overline{s}} \Leftrightarrow \varphi_s$ and $\bigwedge_{i\in I} \varphi_i \Rightarrow \varphi_i$ for any $i \in I$.

The following property can be established by an easy inductive proof.

Lemma 2 For any $\Delta \in \mathcal{D}(\mathsf{sCSP})$ we have $\Delta \models \varphi_\Delta$. □

It and the following lemma help to prove Thm. 4.

Lemma 3 For any processes $P, Q \in \mathsf{pCSP}$ we have that $[\![P]\!] \models \varphi_{[\![Q]\!]}$ implies $P \sqsubseteq_{FS} Q$.

Proof: Define $\mathcal{R}$ by $s\ \mathcal{R}\ \Theta$ iff $\Theta \models \varphi_s$. We first show that

$$\Theta \models \varphi_\Delta \ \text{ implies } \ \exists \Theta' : \Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta' \wedge \Delta\ \overline{\mathcal{R}}\ \Theta'. \qquad (1)$$

Suppose $\Theta \models \varphi_\Delta$ with $\varphi_\Delta = \bigoplus_{i\in I} p_i{\cdot}\varphi_{s_i}$, so for all $i \in I$ there are $\Theta_i \in \mathcal{D}(\mathsf{sCSP})$ with $\Theta_i \models \varphi_{s_i}$ such that $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ with $\Theta' := \bigoplus_{i\in I} p_i{\cdot}\Theta_i$. Since $s_i\ \mathcal{R}\ \Theta_i$ for all $i \in I$ we have $\Delta\ \overline{\mathcal{R}}\ \Theta'$.

Now we show that $\mathcal{R}$ is a failure simulation.

- Suppose $s\ \mathcal{R}\ \Theta$ and $s \stackrel{\tau}{\longrightarrow} \Delta$. Then $\varphi_s \Rightarrow \varphi_\Delta$, so $\Theta \models \varphi_\Delta$. Now apply (1).
- Suppose $s\ \mathcal{R}\ \Theta$ and $s \stackrel{a}{\longrightarrow} \Delta$ with $a \in A$. Then $\varphi_s \Rightarrow \langle a\rangle\varphi_\Delta$, so $\Theta \models \langle a\rangle\varphi_\Delta$. Hence $\exists\Theta'$ with $\Theta \stackrel{\hat{a}}{\Longrightarrow} \Theta'$ and $\Theta' \models \varphi_\Delta$. Now apply (1).
- Suppose $s\ \mathcal{R}\ \Theta$ and $s \stackrel{X}{\not\rightarrow}$ with $X \subseteq A$. Then $\varphi_s \Rightarrow \mathbf{ref}(X)$, so $\Theta \models \mathbf{ref}(X)$. Hence $\exists\Theta'$ with $\Theta \stackrel{\hat{\tau}}{\Longrightarrow} \Theta'$ and $\Theta' \stackrel{X}{\not\rightarrow}$.

Thus we have $\Theta \models \varphi_s$ implies $s \rhd_{FS} \Theta$. Using (1) with $[\![P]\!] \models \varphi_{[\![Q]\!]}$ gives $P \sqsubseteq_{FS} Q$ via Def. 5. □

Theorem 4

1. If $[\![P]\!] \sqsubseteq^{\mathcal{L}^-} [\![Q]\!]$ then $P \sqsubseteq_S Q$.
2. If $[\![P]\!] \sqsubseteq^{\mathcal{L}} [\![Q]\!]$ then $P \sqsubseteq_{FS} Q$.

Proof: Suppose $[\![P]\!] \sqsubseteq^{\mathcal{L}} [\![Q]\!]$. By Lem. 2 we have $[\![Q]\!] \models \varphi_{[\![Q]\!]}$ and hence $[\![P]\!] \models \varphi_{[\![Q]\!]}$. Lem. 3 gives $P \sqsubseteq_{FS} Q$.

For the $\sqsubseteq^{\mathcal{L}^-}$ case, omit $\mathbf{ref}(X)$ from the definition of a characteristic formula and begin with $[\![P]\!] \models \varphi_{[\![P]\!]}$. The counterpart of Lem. 3 now says that $[\![Q]\!] \models \varphi_{[\![P]\!]}$ implies $P \sqsubseteq_S Q$. □

8 Characteristic tests

Our final step towards Thm. 2 is taken in this section, where we show that every modal formula φ can be characterised by a vector-based test T_φ such that any pCSP process satisfies φ just when it passes the test T_φ.

Lemma 4 For every $\varphi \in \mathcal{L}$ there exists a pair (T_φ, v_φ) with T_φ an Ω-test and $v_\varphi \in [0,1]^\Omega$, such that

$$\Delta \models \varphi \quad \Leftrightarrow \quad \exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi \qquad (2)$$

for all $\Delta \in \mathcal{D}(\mathsf{sCSP})$, and in case $\varphi \in \mathcal{L}^-$ we also have

$$\Delta \models \varphi \quad \Leftrightarrow \quad \exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \geq v_\varphi \,. \qquad (3)$$

T_φ is called a *characteristic test* of φ and v_φ its *target value*.

Proof: First of all note that if a pair (T_φ, v_φ) satisfies the requirements above, then any pair obtained from (T_φ, v_φ) by bijectively renaming the elements of Ω also satisfies these requirements. Hence a characteristic test can always be chosen in such a way that there is a success action $\omega \in \Omega$ that does not occur in (the finite) T_φ. Moreover, any countable collection of characteristic tests can be assumed to be Ω-*disjoint*, meaning that no $\omega \in \Omega$ occurs in two different elements of the collection.

The required characteristic tests and target values are obtained as follows.

- Let $\varphi = \top$. Take $T_\varphi := \omega$ for some $\omega \in \Omega$, and $v_\varphi := \vec{\omega}$.
- Let $\varphi = \langle a \rangle \psi$. By induction, ψ has a characteristic test T_ψ with target value v_ψ. Take $T_\varphi := \omega \,\Box\, a.T_\psi$ where $\omega \in \Omega$ does not occur in T_ψ, and $v_\varphi := v_\psi$.
- Let $\varphi = \mathbf{ref}(X)$ with $X \subseteq \mathsf{Act}$. Take $T_\varphi := \Box_{a \in X} \, a.\omega$ for some $\omega \in \Omega$, and $v_\varphi = \vec{0}$.
- Let $\varphi = \bigwedge_{i \in I} \varphi_i$ with I a finite and non-empty index set. Choose a Ω-disjoint family $(T_i, v_i)_{i \in I}$ of characteristic tests T_i with target values v_i for each φ_i. Furthermore, let $p_i \in (0,1]$ for $i \in I$ be chosen arbitrarily such that $\sum_{i \in I} p_i = 1$. Take $T_\varphi := \bigoplus_{i \in I} p_i \cdot T_i$ and $v_\varphi := \sum_{i \in I} p_i v_i$.
- Let $\varphi = \bigoplus_{i \in I} p_i \cdot \varphi_i$. Choose a Ω-disjoint family $(T_i, v_i)_{i \in I}$ of characteristic tests T_i with target values v_i for each φ_i, such that there are distinct success actions ω_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 φ_i with target value v'_i. Take $T_\varphi := \bigsqcap_{i \in I} T'_i$ and $v_\varphi := \sum_{i \in I} p_i v'_i$.

Note that $v_\varphi(\omega) = 0$ whenever $\omega \in \Omega$ does not occur in T_φ. By induction on φ we now check (2) above.

- Let $\varphi = \top$. For all $\Delta \in \mathcal{D}(\mathsf{sCSP})$ we have $\Delta \models \varphi$ as well as $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi$, using Lem. 1.
- Let $\varphi = \langle a \rangle \psi$ with $a \in \mathsf{Act}$. Suppose $\Delta \models \varphi$. Then there is a Δ' with $\Delta \stackrel{\hat{a}}{\Longrightarrow} \Delta'$ and $\Delta' \models \psi$. By induction, $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\psi, \Delta') : \; o \leq v_\psi$. By Lem. 1, $o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta)$.

  Now suppose $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi$. This implies $o(\omega) = 0$, so by Lem. 1 there is a Δ' with $\Delta \stackrel{\hat{a}}{\Longrightarrow} \Delta'$ and $o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\psi, \Delta')$. By induction, $\Delta' \models \psi$, so $\Delta \models \varphi$.
- Let $\varphi = \mathbf{ref}(X)$ with $X \subseteq \mathsf{Act}$. Suppose $\Delta \models \varphi$. Then there is a Δ' with $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \stackrel{X}{\not\longrightarrow}$. By Lem. 1, $\vec{0} \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta)$.

  Now suppose $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi$. This implies $o = \vec{0}$, so by Lem. 1 there is a Δ' with $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$ and $\Delta' \stackrel{X}{\not\longrightarrow}$. Hence $\Delta \models \varphi$.
- Let $\varphi = \bigwedge_{i \in I} \varphi_i$ with I a finite and non-empty index set. Suppose $\Delta \models \varphi$. Then $\Delta \models \varphi_i$ for all $i \in I$, and hence, by induction, $\exists o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_i, \Delta) : \; o_i \leq v_i$. Thus $o := \sum_{i \in I} p_i o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta)$ by Lem. 1, and $o \leq v_\varphi$.

  Now suppose $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi$. Then, using Lem. 1, $o = \sum_{i \in I} p_i o_i$ for certain $o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_i, \Delta)$. One has $o_i \leq v_i$ for all $i \in I$, for if $o_i(\omega) > v_i(\omega)$ for some $i \in I$ and $\omega \in \Omega$, then ω must occur in T_i and hence cannot occur in T_j for $j \neq i$. This implies $o_j(\omega) = 0$ for all $j \neq i$ and thus $o(\omega) > v_\varphi(\omega)$, in contradiction with the assumption. By induction, $\Delta \models \varphi_i$ for all $i \in I$, and hence $\Delta \models \varphi$.
- Let $\varphi = \bigoplus_{i \in I} p_i \cdot \varphi_i$. Suppose $\Delta \models \varphi$. Then for all $i \in I$ there are $\Delta_i \in \mathcal{D}(\mathsf{sCSP})$ with $\Delta_i \models \varphi_i$ such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \bigoplus_{i \in I} p_i \cdot \Delta_i$. By induction, there are $o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(\Delta_i, T'_i)$ with $o_i \leq v'_i$. By Lem. 1, $o := \sum_{i \in I} p_i o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta)$, and $o \leq v_\varphi$.

  Now suppose $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta) : \; o \leq v_\varphi$. Then, by Lem. 1, there are $q \in \mathcal{D}(I)$ and Δ_i, for $i \in I$, such that $\Delta \stackrel{\hat{\tau}}{\Longrightarrow} \bigoplus_{i \in I} q_i \cdot \Delta_i$ and $o = \sum_{i \in I} q_i o_i$ for some $o_i \in \widehat{\mathcal{A}}^\Omega_\updownarrow(\Delta_i, T'_i)$. Now $\forall i : o_i(\omega_i) = v'_i(\omega_i) = \frac{1}{2}$, so $\frac{1}{2} q_i = q_i o_i(\omega_i) = o(\omega_i) \leq v_\varphi(\omega_i) = p_i v'_i(\omega_i) = \frac{1}{2} p_i$. 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 one obtains $o_i \leq v'_i$ all $i \in I$. By induction, $\Delta_i \models \varphi_i$ for all $i \in I$, and hence $\Delta \models \varphi$.

In case $\varphi \in \mathcal{L}^-$, a straightforward induction yields that $|v_\varphi| = 1$ and for all $\Delta \in \mathcal{D}(\mathsf{pCSP})$ and $o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, \Delta)$ we have $|o| = 1$. Here $|o|$ denotes $\sum_{\omega \in \Omega} o(\omega)$. Therefore, $o \leq v$ iff $o \geq v$ iff $o = v$, yielding (3). $\Box$

Theorem 5

1. If $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmay}}\, Q$ then $[\![P]\!] \sqsubseteq^{\mathcal{L}^-} [\![Q]\!]$.
2. If $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q$ then $[\![P]\!] \sqsubseteq^{\mathcal{L}} [\![Q]\!]$.

Proof: Suppose $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q$ and $[\![Q]\!] \models \varphi$ for some $\varphi \in \mathcal{L}$. Let T_φ be a characteristic test of φ with target value v_φ. Then Lem. 4 yields $\exists o \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, [\![Q]\!])$. $o \leq v_\varphi$, and hence, given that $P \,\widehat{\sqsubseteq}^\Omega_{\text{pmust}}\, Q$ and $\widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, [\![R]\!]) = \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, R)$ for any $R \in \mathsf{pCSP}$, we have $\exists o' \in \widehat{\mathcal{A}}^\Omega_\updownarrow(T_\varphi, [\![P]\!]) : \; o' \leq v_\varphi$. Thus $[\![P]\!] \models \varphi$.

The may-case goes likewise. $\Box$

$$
\begin{array}{lrl}
(\mathbf{P1}) & P \,{}_p\!\oplus P & = P \\
(\mathbf{P2}) & P \,{}_p\!\oplus Q & = Q \,{}_{1-p}\!\oplus P \\
(\mathbf{P3}) & (P \,{}_p\!\oplus Q) \,{}_q\!\oplus R & = P \,{}_{p\cdot q}\!\oplus (Q \,{}_{\frac{(1-p)\cdot q}{1-p\cdot q}}\!\oplus R) \\
(\mathbf{I1}) & P \sqcap P & = P \\
(\mathbf{I2}) & P \sqcap Q & = Q \sqcap P \\
(\mathbf{I3}) & (P \sqcap Q) \sqcap R & = P \sqcap (Q \sqcap R) \\
(\mathbf{E1}) & P \,\Box\, \mathbf{0} & = P \\
(\mathbf{E2}) & P \,\Box\, Q & = Q \,\Box\, P \\
(\mathbf{E3}) & (P \,\Box\, Q) \,\Box\, R & = P \,\Box\, (Q \,\Box\, R) \\
(\mathbf{EI}) & a.P \,\Box\, a.Q & = a.P \sqcap a.Q \\
(\mathbf{D1}) & P \,\Box\, (Q \,{}_p\!\oplus R) & = (P \,\Box\, Q) \,{}_p\!\oplus (P \,\Box\, R) \\
(\mathbf{D2}) & a.P \,\Box\, (Q \sqcap R) & = (a.P \,\Box\, Q) \sqcap (a.P \,\Box\, R) \\
(\mathbf{D3}) & P \,\Box\, Q & = (P_1 \,\Box\, Q) \sqcap (P_2 \,\Box\, Q) \\
 & & \quad \sqcap (P \,\Box\, Q_1) \sqcap (P \,\Box\, Q_2), \\
 & \text{provided} & P = P_1 \sqcap P_2,\ Q = Q_1 \sqcap Q_2
\end{array}
$$

Figure 2. Common equations

Combining Thms. 3-5 we obtain Thm. 2, the goal we set ourselves in Sec. 5. Thus, with Thm. 1 and Prop. 2, we have shown that the may preorder coincides with simulation and that the must preorder coincides with failure simulation.

9 Equational theories

In order to focus on the essentials let us only consider processes which do not use the parallel operator $|_A$; we call the sub-language nCSP. For a discussion of the axiomatisation for terms involving $|_A$, and the other parallel operators commonly used in CSP see Sec. 12.

Let us write $P =_E Q$ to denote that $P = Q$ can be derived using the equations given in Fig. 2. (Given the way we defined the syntax of pCSP, axiom $(\mathbf{D1})$ is merely a case of abbreviation-expansion.) Many of the standard equations for CSP [15] are missing; typical examples include:

$$
\begin{array}{c}
a.(P \sqcap Q) = a.P \sqcap a.Q \\
P = P \,\Box\, P \\
P \,\Box\, (Q \sqcap R) = (P \,\Box\, Q) \sqcap (P \,\Box\, R) \\
P \sqcap (Q \,\Box\, R) = (P \sqcap Q) \,\Box\, (P \sqcap R)
\end{array}
$$

For a detailed discussion of the standard equations for CSP in the presence of probabilistic processes see Sect. 4 of [9].

Proposition 3 *Suppose $P =_E Q$. Then $P \simeq_{FS} Q$.*

Proof: Because of Prop. 1 it is sufficient to exhibit witness failure simulations for axioms in Fig. 2. □

Note that this result also means $P =_E Q$ implies $P \simeq_S Q$.

Despite the weakness of this equational theory, it does allow us to reduce terms to a form in which the external choice operator is only applied to prefix terms.

May:

$$
\begin{array}{lrl}
(\mathbf{May0}) & a.P \,\Box\, b.Q & = a.P \sqcap b.Q \\
(\mathbf{May1}) & P & \sqsubseteq P \sqcap Q \\
(\mathbf{May2}) & \mathbf{0} & \sqsubseteq P \\
(\mathbf{May3}) & a.(P \,{}_p\!\oplus Q) & \sqsubseteq a.P \,{}_p\!\oplus a.Q
\end{array}
$$

Must:

$$
\begin{array}{lrl}
(\mathbf{Must1}) & P \sqcap Q & \sqsubseteq Q \\
(\mathbf{Must2}) & R \sqcap \bigsqcap_{i\in I} P_i & \sqsubseteq \bigsqcap_{i\in I} a_i.Q_i, \\
 & \text{provided} & P_i = \bigoplus_{j\in J_i} p_j (a_i.Q_{ij} \,\Box\, P_{ij}) \\
 & & Q_i = \bigoplus_{j\in J_i} p_j Q_{ij} \\
 & & \mathit{inits}(R) \subseteq \{a_i\}_{i\in I}
\end{array}
$$

Figure 3. Inequations

Definition 8 [Head standard forms] The set of *head standard forms H* is given by the following grammar:

$$
\begin{array}{rcl}
H & ::= & \bigoplus_{i\in I} p_i A_i \\
A & ::= & \Box_{i\in I} a_i.P_i \ \mid\ H_1 \sqcap H_2
\end{array}
$$

Proposition 4 *For every $P \in$ nCSP there is a head standard form $H(P)$ such that*

- $P =_E H(P)$
- $|P| = |H(P)|$

Here we use $|P|$ to denote the depth *of P, the length of the largest path in the pLTS $[\![P]\!]$.*

Proof: A straightforward induction, heavily relying on $(\mathbf{D1})$–$(\mathbf{D3})$. □

We can also show that the axioms $(\mathbf{P1})$–$(\mathbf{P3})$ are in some sense all that are required to reason about probabilistic choice. Let $P =_{\text{prob}} Q$ denote the associated equational theory. Then we have the following property.

Lemma 5 Suppose P, Q take the form $\bigoplus_{i\in I} S_i$ and $\bigoplus_{j\in J} T_j$ respectively, where each S_i, T_j are state-based processes. Then $[\![P]\!] = [\![Q]\!]$ implies $P =_{\text{prob}} Q$. □

10 Inequational theories

In order to characterise the simulation preorders, and the associated testing preorders, we introduce *inequations*. We write $P \sqsubseteq_{E_{\text{may}}} Q$ when $P \sqsubseteq Q$ is derivable from the inequational theory obtained by adding the four *may* inequations in Fig. 3 to the equations in Fig. 2. Recall that the first three

additions, ($\mathbf{May0}$)–($\mathbf{May2}$), are used in the standard testing theory of CSP [15, 7]. For the *must* case, in addition to the standard inequation ($\mathbf{Must1}$), we require an inequational schema, ($\mathbf{Must2}$); this uses the notation *inits*(P) to denote the (finite) set of initial visible actions of P. Formally,

$$\begin{array}{rcl} \mathit{inits}(\bigoplus_{i\in I} p_i A_i) & = & \bigcup_{i\in I}\{\mathit{inits}(A_i)\} \\ \mathit{inits}(\sum_{i\in I} a_i P_i) & = & \{a_i\}_{i\in I} \\ \mathit{inits}(H_1 \sqcap H_2) & = & \mathit{inits}(H_1) \cup \mathit{inits}(H_2) \end{array}$$

The side conditions of ($\mathbf{Must2}$) entail that $P_i \xrightarrow{a_i} [\![Q_i]\!]$ and there exists some Δ such that $R \stackrel{\hat{\tau}}{\Longrightarrow} \Delta \stackrel{X}{\not\longrightarrow}$ with $X = \mathsf{Act}\backslash\{a_i\}_{i\in I}$. Note that ($\mathbf{Must2}$) can be used, together with ($\mathbf{I1}$), to derive the dual of ($\mathbf{May3}$): $a.P \,{}_p\!\oplus\, a.Q \sqsubseteq a.(P \,{}_p\!\oplus\, Q)$. We write $P \sqsubseteq_{E_{\text{must}}} Q$ when $P \sqsubseteq Q$ is derivable from the resulting inequational theory.

Theorem 6 For P, Q in nCSP, it holds that

(i) $P \sqsubseteq_S Q$ if and only if $P \sqsubseteq_{E_{\text{may}}} Q$

(ii) $P \sqsubseteq_{FS} Q$ if and only if $P \sqsubseteq_{E_{\text{must}}} Q$

Proof: For one direction it is sufficient to check that the inequations, and the inequational scheme in Fig. 3 are sound. The converse, completeness, is outlined in the next section; it is a combination of Props. 5 and 6. □

An important inequation that follows from ($\mathbf{May1}$) and ($\mathbf{P1}$) is

$$(\mathbf{May4}) \quad P \,{}_p\!\oplus\, Q \sqsubseteq_{E_{\text{may}}} P \sqcap Q$$

saying that any probabilistic choice can be simulated by an internal choice.

11 Completeness

The proof of Thm. 6 depends on a number of properties of the simulation preorders, the most important of which is the following variation on the *Derivative lemma* of [25]:

Lemma 6 [Derivative lemma]

(i) If $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$ then there exists some Q such that $[\![Q]\!] = \Delta$ and (a) $P \sqsubseteq_{E_{\text{must}}} Q$, (b) $Q \sqsubseteq_{E_{\text{may}}} P$.

(ii) If $[\![P]\!] \stackrel{a}{\Longrightarrow} \Delta$ then there exists some Q such that $[\![Q]\!] = \Delta$ and $a.Q \sqsubseteq_{E_{\text{may}}} P$.

Proof: The proof of (i) proceeds in two stages. First suppose $[\![P]\!] \xrightarrow{\hat{\tau}} \Delta$. We use structural induction on P to show that the requirements on Δ are satisfied.

The more general case, when $[\![P]\!] \xrightarrow{\hat{\tau}}{}^* \Delta$, is now a simple inductive argument on the length of the derivation.

The proof of (ii) is similar; first we treat the case when $[\![P]\!] \xrightarrow{a} \Delta$, by structural induction, and then use part (i) to derive the more general case. □

The completeness result now follows from the following two results.

Proposition 5 *$P \sqsubseteq_S Q$ implies $P \sqsubseteq_{E_{may}} Q$.*

Proof: See Appendix A. □

Proposition 6 *$P \sqsubseteq_{FS} Q$ implies $P \sqsubseteq_{E_{must}} Q$.*

Proof: See Appendix B. □

12 Conclusions

In this paper we continue our previous work [9, 10] in our quest for a testing theory for processes which exhibit both nondeterministic and probabilistic behaviour. We have studied may- and must testing preorders for finite processes from three different aspects: (i) we have shown that the may preorder can be characterised as a co-inductive simulation relation, and the must preorder as a failure simulation relation; (ii) we have given a characterisation of both preorders in a finitary modal logic; and (iii) we have also provided complete axiomatisations for both preorders over a probabilistic version of CSP. Although we omitted our parallel operator $|_A$ from the axiomatisations, it and similar CSP and CCS-like parallel operators can be handled using standard techniques, in the must case at the expense of introducing auxiliary operators. In future work we hope to extend these results to finitely branching and finite-state processes.

We believe these results, in each of the three areas, to be novel, although a number of partial results along similar lines already exist in the literature. These are detailed below.

Related work: Probabilistic extensions of testing equivalences [7] have been widely studied. There are two different proposals on how to set up a test: (i) a test should be nonprobabilistic, i.e., there is no occurrence of probabilistic choice in a test (e.g. [2, 5, 21, 16, 20, 11]); or (ii) a test can be probabilistic, i.e., probabilistic choice may occur in processes as well as tests (e.g. [32, 39, 17, 18, 30, 4]). This paper adopts the second approach.

Some work [32, 30] does not consider nondeterminism but deals exclusively with fully probabilistic processes. In this setting a process passes a test with a unique probability instead of a set of probabilities. For this kind of processes, Cleaveland et al. defined their testing preorders [32] and characterised them in terms of *probabilistic traces* [40, 6]; Núñez et al. defined a testing equivalence and gave alternative characterisations in terms of *probabilistic acceptance trees* and sets of axioms [30, 29]. It would be interesting to see if denotational characterisations of our testing preorders can be done by using ideas similar to the above mentioned work.

Cazorla et al. [4] considered nondeterminism and extended the results of [30, 29] to a probabilistic process algebra with internal, external, and probabilistic choices. Since they chose to resolve probabilistic choice before internal and external choices, some properties of classical process algebras are not maintained. For instance, both internal and external choices fail to be idempotent. The same problem was encountered in e.g. [39, 38, 28] as well. This problem was addressed in [26] where Mislove gave a denotational model to justify some intuitively correct laws of internal and probabilistic choices. However, he did not consider other CSP laws. For example, since the external choice operator $\Box$ has been altered, it is unclear whether the law $a.P \,\Box\, a.Q = a.P \sqcap a.Q$ holds in his model. In our work we do not allow probabilistic choice to be resolved before internal choice in processes of the form $P \sqcap Q$, so internal choice remains idempotent, though external choice does not.

The work most closely related to ours is [17, 18], where Jonsson and Wang characterised a may preorder using their notion of simulation which is weaker than $\sqsubseteq_S$ (cf. Def. 5). However, they have only considered processes without τ-moves. In [9] we have shown that tests with internal moves can distinguish more processes than tests without internal moves, even when applied to processes that have no internal moves themselves.

Segala [34] defined two preorders called trace distribution precongruence ($\sqsubseteq_{TD}$) and failure distribution precongruence ($\sqsubseteq_{FD}$). He has proved that the former coincides with $\widehat{\sqsubseteq}^{\Omega}_{\text{pmay}}$ (cf. Def. 6) and that the latter coincides with $\widehat{\sqsubseteq}^{\Omega}_{\text{pmust}}$. In [24] it has been shown that $\sqsubseteq_{TD}$ coincides with a notion of simulation similar to $\sqsubseteq_S$. It follows from [34] and [24] that $\widehat{\sqsubseteq}^{\Omega}_{\text{pmay}}$ coincides with their simulation. It is tempting to establish similar link between $\widehat{\sqsubseteq}^{\Omega}_{\text{pmust}}$ and failure simulation. However, the proof technique used in [24] is very involved and tailored to simulation. It is unclear how to apply it to relate $\sqsubseteq_{FD}$ to failure simulation. In contrast, our approach of using modal characterisations is very concise, and it works for both may- and must preorders.

Some work [19, 22, 36] defines probabilistic equivalences based on traces, failures and readies. These equivalences are even coarser than $\simeq_S$. For example, let

$$P := a.((b.d \,\Box\, c.e) \,{}_{\frac{1}{2}}\!\oplus\, (b.f \,\Box\, c.g))$$
$$Q := a.((b.d \,\Box\, c.g) \,{}_{\frac{1}{2}}\!\oplus\, (b.f \,\Box\, c.e)).$$

The two processes cannot be distinguished by the equivalences of [19, 22, 36]. However, we can tell them apart by the test

$$T := a.((b.d.\omega \,{}_{\frac{1}{2}}\!\oplus\, c.e.\omega) \sqcap (b.f.\omega \,{}_{\frac{1}{2}}\!\oplus\, c.g.\omega))$$

since $\mathcal{A}(T,P) = \{0, \frac{1}{2}, 1\}$ and $\mathcal{A}(T,Q) = \{\frac{1}{2}\}$, that is, $P \not\sqsubseteq_{\text{pmay}} Q$.

Finally, there is an extensive literature on probabilistic extensions of simulation; the reader is refered to [9] for a detailed account.

References

[1] E. Bandini and R. Segala. Axiomatizations for probabilistic bisimulation. In Proc. of *ICALP'01*, volume 2076 of LNCS, pages 370–381. Springer, 2001.

[2] B. Bloom and A.R. Meyer. A remark on bisimulation between probabilistic processes. In Proc. of *Logic at Botik '89*, volume 363 of LNCS, pages 26–40. Springer, 1989.

[3] S.D. Brookes, C.A.R. Hoare, and A.W. Roscoe. A theory of communicating sequential processes. *Journal of the ACM*, 31(3):560–599, 1984.

[4] D. Cazorla, F. Cuartero, V. V. Ruiz, F. L. Pelayo, and J.J. Pardo. Algebraic theory of probabilistic and nondeterministic processes. *Journal of Logic and Algebraic Programming*, 55(1-2):57–103, 2003.

[5] I. Christoff. Testing equivalences and fully abstract models for probabilistic processes. In Proc. of *CONCUR'90*, volume 458 of LNCS, pages 126–140. Springer, 1990.

[6] R. Cleaveland, Z. Dayar, S. A. Smolka, and S. Yuen. Testing preorders for probabilistic processes. *Information and Computation*, 154(2):93–148, 1999.

[7] R. De Nicola and M. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.

[8] Y. Deng and C. Palamidessi. Axiomatizations for probabilistic finite-state behaviors. In Proc. of *FOSSACS'05*, volume 3441 of LNCS, pages 110–124. Springer, 2005.

[9] Y. Deng, R. van Glabbeek, M. Hennessy, C. Morgan, and C. Zhang. Remarks on testing probabilistic processes. *ENTCS*, 2007. To appear. Available at `http://www.cse.unsw.edu.au/~yuxind/publications/gdpf.pdf`.

[10] Y. Deng, R. van Glabbeek, C. Morgan, and C. Zhang. Scalar outcomes suffice for finitary probabilistic testing. In Proc. of *ESOP'07*, LNCS. Springer, 2007. To appear. Available at `http://www.cse.unsw.edu.au/~yuxind/publications/scalar.pdf`.

[11] C. Gregorio-Rodríguez and M. Núñez. Denotational semantics for probabilistic refusal testing. *ENTCS*, 22, 1999.

[12] H. Hansson and B. Jonsson. A calculus for communicating systems with time and probabilities. In Proc. of *RTSS'90*, pages 278–287. IEEE Computer Society Press, 1990.

[13] Jifeng He, K. Seidel, and A.K. McIver. Probabilistic models for the guarded command language. *Science of Computer Programming*, 28:171–192, 1997.

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

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

[16] B. Jonsson, C. Ho-Stuart, and Wang Yi. Testing and refinement for nondeterministic and probabilistic processes. In Proc. of *FTRTFT'94*, volume 863 of LNCS, pages 418–430. Springer, 1994.

[17] B. Jonsson and Wang Yi. Compositional testing preorders for probabilistic processes. In Proc. of *LICS'95*, pages 431–441. IEEE Computer Society, 1995.

[18] B. Jonsson and Wang Yi. Testing preorders for probabilistic processes can be characterized by simulations. *Theoretical Computer Science*, 282(1):33–51, 2002.

[19] C. Jou and S.A. Smolka. Equivalences, congruences, and complete axiomatizations for probabilistic processes. In Proc. of *CONCUR '90*, volume 458 of LNCS, pages 367–383. Springer, 1990.

[20] M.Z. Kwiatkowska and G. Norman. A testing equivalence for reactive probabilistic processes. *ENTCS*, 16(2), 1998.

[21] K.G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94(1):1–28, 1991.

[22] G. Lowe. Representing nondeterminism and probabilistic behaviour in reactive processes. Technical Report TR-11-93, Computing laboratory, Oxford University, 1993.

[23] G. Lowe. Probabilistic and prioritized models of timed CSP. *Theoretical Computer Science*, 138:315–352, 1995.

[24] N. Lynch, R. Segala, and F.W. Vaandrager. Compositionality for probabilistic automata. In Proc. of *CONCUR'03*, volume 2761 of LNCS, pages 204–222. Springer, 2003.

[25] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[26] M.W. Mislove. Nondeterminism and probabilistic choice: Obeying the laws. In Proc. of *CONCUR'00*, volume 1877 of LNCS, pages 350–364. Springer, 2000.

[27] M.W. Mislove, J. Ouaknine, and J. Worrell. Axioms for probability and nondeterminism. *ENTCS*, 96:7–28, 2004.

[28] C. Morgan, A. McIver, K. Seidel, and J.W. Sanders. Refinement oriented probability for CSP. *Formal Aspects of Computing*, 8:617–647, 1996.

[29] M. Núñez. Algebraic theory of probabilistic processes. *Journal of Logic and Algebraic Programming*, 56(1-2):117–177, 2003.

[30] M. Núñez, D. de Frutos-Escrig, and L.F.L. Díaz. Acceptance trees for probabilistic processes. In Proc. of *CONCUR '95*, volume 962 of LNCS, pages 249–263. Springer, 1995.

[31] E.-R. Olderog and C.A.R. Hoare. Specification-oriented semantics for communicating processes. *Acta Informatica*, 23:9–66, 1986.

[32] S.A. Smolka R. Cleaveland and A.E. Zwarico. Testing preorders for probabilistic processes. In Proc. of *ICALP'92*, volume 623 of LNCS, pages 708–719. Springer, 1992.

[33] R. Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, MIT, 1995.

[34] R. Segala. Testing probabilistic automata. In Proc. of *CONCUR'96*, volume 1119 of LNCS, pages 299–314. Springer, 1996.

[35] R. Segala and N.A. Lynch. Probabilistic simulations for probabilistic processes. In Proc. of *CONCUR'94*, volume 836 of LNCS, pages 481–496. Springer, 1994.

[36] K. Seidel. Probabilistic communicating processes. *Theoretical Computer Science*, 152:219–249, 1995.

[37] R. Tix, K. Keimel, and G.D. Plotkin. Semantic domains for combining probability and non-determinism. *ENTCS*, 129:1–104, 2005.

[38] Wang Yi. Algebraic reasoning for real-time probabilistic processes with uncertain information. In Proc. of *FTRTFT'94*, volume 863 of LNCS, pages 680–693. Springer, 1994.

[39] Wang Yi and K.G. Larsen. Testing probabilistic and nondeterministic processes. In Proc. IFIP TC6/WG6.1 12th Int. Symp. on *Protocol Specification, Testing and Verification*, volume C-8 of *IFIP Transactions*, pages 47–61. North-Holland, 1992.

[40] S. Yuen, R. Cleaveland, Z. Dayar, and S. A. Smolka. Fully abstract characterizations of testing preorders for probabilistic processes. In Proc. of *CONCUR '94*, volume 836 of LNCS, pages 497–512. Springer, 1994.

A Proof of Prop. 5

First we state without proof the following properties:

Lemma 7 For $\rhd \in \{\rhd_{\mathcal{S}}, \rhd_{FS}\}$

(i) $\overline{s} \,\overline{\rhd}\, \Delta$ if and only if $s \rhd \Delta$

(ii) $\Delta \,\overline{\rhd}\, \overline{t}$ if and only if $s \rhd \overline{t}$ for every s in the support of Δ. □

Proof (of Prop. 5) The proof is by induction on the combined size of P, Q, and we may assume that both are in head standard form. We do a case analysis on the structure of P. There are three cases.

(i) P has the form $\Box_{i\in I}\, a_i.P_i$. If I contains two or more elements then P may also be written as $\sqcap_{i\in I}\ a_i.P_i$, using $(\mathbf{May0})$, and we may proceed as in case (ii) below. If I is empty, that is P is $\mathbf{0}$, then we can use $(\mathbf{May2})$. So we are left with the possibility that P is $a.P'$. Suppose $a.P' \sqsubseteq_S Q$, i.e. $[\![a.P']\!] \,\overline{\rhd_{\mathcal{S}}}\, \Delta$ for some $[\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$. From the Derivative Lemma we know there is some Q' such that $[\![Q']\!] = \Delta$ and $Q' \sqsubseteq_{E_{\text{may}}} Q$. So it suffices to show $a.P' \sqsubseteq_{E_{\text{may}}} Q'$, which we do by analysing the structure of Q'.

- Q' is $a.Q''$. We know from $[\![a.P']\!] \,\overline{\rhd_{\mathcal{S}}}\, [\![a.Q'']\!]$ that $[\![P']\!] \,\overline{\rhd_{\mathcal{S}}}\, \Delta'$ for some $[\![Q'']\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta'$, thus $P' \sqsubseteq_S Q''$. Therefore, we have $P' \sqsubseteq_{E_{\text{may}}} Q''$ by induction. It follows that $a.P' \sqsubseteq_{E_{\text{may}}} a.Q''$.
- Q' is $\Box_{i\in I}\, a_i.P_i$ with at least two elements in I. We use $(\mathbf{May0})$ and then proceed as in the next case.
- Q' is $Q_1 \sqcap Q_2$. We know from $[\![a.P']\!] \,\overline{\rhd_{\mathcal{S}}}\, [\![Q_1 \sqcap Q_2]\!]$ that $[\![P']\!] \,\overline{\rhd_{\mathcal{S}}}\, \Delta'$ for some Δ' such that one of the following two conditions holds
  (a) $[\![Q_i]\!] \stackrel{a}{\Longrightarrow} \Delta'$ for $i = 1$ or 2. In this case, $a.P' \sqsubseteq_S Q_i$. By induction we have $a.P' \sqsubseteq_{E_{\text{may}}} Q_i$, then we apply $(\mathbf{May1})$.
  (b) $[\![Q_1]\!] \stackrel{a}{\Longrightarrow} \Delta_1$ and $[\![Q_2]\!] \stackrel{a}{\Longrightarrow} \Delta_2$ such that $\Delta' = \Delta_1 \,{}_p\!\oplus\, \Delta_2$. By the Derivative Lemma, there exist Q_1', Q_2' such that $[\![Q_1']\!] = \Delta_1$, $[\![Q_2']\!] = \Delta_2$, $a.Q_1' \sqsubseteq_{E_{\text{may}}} Q_1$ and $a.Q_2' \sqsubseteq_{E_{\text{may}}} Q_2$. Clearly, $[\![Q_1' \,{}_p\!\oplus\, Q_2']\!] = \Delta'$, thus $P' \sqsubseteq_S Q_1' \,{}_p\!\oplus\, Q_2'$. By induction, we have $P' \sqsubseteq_{E_{\text{may}}} Q_1' \,{}_p\!\oplus\, Q_2'$. So

$$\begin{aligned} a.P' &\sqsubseteq_{E_{\text{may}}} a.(Q_1' \,{}_p\!\oplus\, Q_2') \\ &\sqsubseteq_{E_{\text{may}}} a.Q_1' \,{}_p\!\oplus\, a.Q_2' \qquad (\mathbf{May3}) \\ &\sqsubseteq_{E_{\text{may}}} Q_1 \,{}_p\!\oplus\, Q_2 \\ &\sqsubseteq_{E_{\text{may}}} Q_1 \sqcap Q_2 \qquad (\mathbf{May4}) \end{aligned}$$

- Q' is $\bigoplus_{i\in I} p_i A_i$. In fact let us write this as $Q_1 \,{}_p\!\oplus\, Q_2$. We know from $[\![a.P']\!] \,\overline{\rhd_{\mathcal{S}}}\, [\![Q_1 \,{}_p\!\oplus\, Q_2]\!]$ that $[\![P']\!] \,\overline{\rhd_{\mathcal{S}}}\, \Delta'$ for some Δ' such that $[\![Q_1 \,{}_p\!\oplus\, Q_2]\!] \stackrel{a}{\Longrightarrow} \Delta'$. From [9, Prop. 6.1 (ii)] and the Derivative Lemma we know that Δ' must take the form $[\![Q_1']\!] \,{}_p\!\oplus\, [\![Q_2']\!]$, where $[\![Q_i]\!] \stackrel{a}{\Longrightarrow} [\![Q_i']\!]$ for $i = 1, 2$. Hence, $P' \sqsubseteq_S Q_1' \,{}_p\!\oplus\, Q_2'$ and by induction we get $P' \sqsubseteq_{E_{\text{may}}} Q_1' \,{}_p\!\oplus\, Q_2'$. Then we can derive $a.P' \sqsubseteq_{E_{\text{may}}} Q_1 \,{}_p\!\oplus\, Q_2$ as in last case.

(ii) P has the form $P_1 \sqcap P_2$. Since $P_i \sqsubseteq_{E_{\text{may}}} P$ we know $P_i \sqsubseteq_S Q$. We use induction to obtain $P_i \sqsubseteq_{E_{\text{may}}} Q$, from which the result follows using $(\mathbf{I1})$.

(iii) P has the form $\bigoplus_{i\in I} p_i A_i$. Suppose $P \sqsubseteq_S Q$, i.e. $[\![P]\!] \,\overline{\rhd_{\mathcal{S}}}\, \Delta$ for some $[\![Q]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta$. From the Derivative Lemma we know there is some Q' such that $[\![Q']\!] = \Delta$ and $Q' \sqsubseteq_{E_{\text{may}}} Q$. So it suffices to show $P \sqsubseteq_{E_{\text{may}}} Q'$, which we do by analysing the structure of Q'.

- Q' takes one of the forms $\Box_{i\in I}\, a_i.Q_i$ or $Q_1 \sqcap Q_2$. Then $[\![Q']\!]$ is a point distribution and we know from Lem. 7 that $[\![A_i]\!] \,\overline{\rhd_{\mathcal{S}}}\, [\![Q']\!]$. So $A_i \sqsubseteq_S Q'$ and by induction we have $A_i \sqsubseteq_{E_{\text{may}}} Q'$. Then $P \sqsubseteq_{E_{\text{may}}} Q'$ follows from $(\mathbf{P_1})$.
- Q' has the form $\bigoplus_{j\in J} q_j B_j$. We have

$$[\![P]\!] = \bigoplus_{k\in K} r_k \cdot \overline{s_k} \quad s_k \rhd_{\mathcal{S}} \Delta_k \quad [\![Q']\!] = \bigoplus_{k\in K} r_k \cdot \Delta_k$$

for some K, r_k, s_k and Δ_k. Now the support of each Δ_k must be contained in that of $[\![Q']\!]$, so we can find a head standard form C_k such that $[\![C_k]\!] = \Delta_k$. Also each s_k must be of the form A_{i_k} for some index $i_k \in I$. Then

(a) $[\![P]\!] = [\![\bigoplus_{k\in K} r_k A_{i_k}]\!]$. By Lem. 5 we have $P =_{\text{prob}} \bigoplus_{k\in K} r_k A_{i_k}$.

(b) $[\![Q']\!] = [\![\bigoplus_{k\in K} r_k C_k]\!]$. Again we can use Lem. 5 to derive that $Q' =_{\text{prob}} \bigoplus_{k\in K} r_k C_k$.

(c) $A_{i_k} \rhd_{\mathcal{S}} [\![C_k]\!]$ implies $A_{i_k} \sqsubseteq_S C_k$, so by induction, $A_{i_k} \sqsubseteq_{E_{\text{may}}} C_k$. Therefore, $\bigoplus_{k\in K} r_k A_{i_k} \sqsubseteq_{E_{\text{may}}} \bigoplus_{k\in K} r_k C_k$.

Combining (a), (b), (c) we obtain $P \sqsubseteq_{E_{\text{may}}} Q'$.

□

B Proof of Prop. 6

Proof: Similar to the proof of Prop. 5, but using a reversed orientation of the preorders. The only real difference is the case when Q has the form $\Box_{i\in I}\, a_i.Q_i$, which we consider now. Let us assume $[\![Q]\!] \,\overline{\rhd_{FS}}\, [\![P]\!]$; if $[\![Q]\!] \,\overline{\rhd_{FS}}\, [\![P']\!]$ for some τ-derivative P' of P, the result will follow in a similar manner, applying the Derivative lemma. By Lem. 7(i) we have $\Box_{i\in I}\, a_i.Q_i \rhd_{FS} [\![P]\!]$. Let X be any set of actions such that $X \cap \{a_i\}_{i\in I} = \emptyset$, then $\Box_{i\in I}\, a_i.Q_i \stackrel{X}{\not\longrightarrow}$. Therefore, there exists some Δ such that $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta \stackrel{X}{\not\longrightarrow}$. By the Derivative lemma, $\Delta = [\![P']\!]$ for some P' such that

$$P \sqsubseteq_{E_{\text{must}}} P' \tag{4}$$

Also, we have $\mathit{inits}(P') \subseteq \{a_i\}_{i\in I}$. Since $\bigsqcap_{i\in I} a_i.Q_i \xrightarrow{a_i} [\![Q_i]\!]$, there exist some Δ_i, Δ_i', Δ_i'' such that $[\![P]\!] \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_i \xrightarrow{a_i} \Delta_i' \stackrel{\hat{\tau}}{\Longrightarrow} \Delta_i''$ and $[\![Q_i]\!] \overline{\rhd_{FS}} \Delta_i''$. Then there exist P_i, P_i', P_i'' such that $\Delta_i = [\![P_i]\!]$, $\Delta_i' = [\![P_i']\!]$, $\Delta_i'' = [\![P_i'']\!]$ and

$$P \sqsubseteq_{E_{\text{must}}} P_i \qquad P_i' \sqsubseteq_{E_{\text{must}}} P_i'' \tag{5}$$

using the Derivative lemma. We know from $[\![P_i]\!] \xrightarrow{a_i} [\![P_i']\!]$ that P_i must be in the form $\bigoplus_{j\in J_i} p_j(a_i.R_{ij} \;\Box\; P_{ij})$ and P_i' in the form $\bigoplus_{j\in J_i} p_j R_{ij}$ for some J_i, R_{ij} and P_{ij}. It can also be seen that $P_i'' \sqsubseteq_{FS} Q_i$. By induction, we have $P_i'' \sqsubseteq_{E_{\text{must}}} Q_i$. It follows that

$$\bigsqcap_{i\in I} a_i.P_i'' \sqsubseteq_{E_{\text{must}}} \bigsqcap_{i\in I} a_i.Q_i \tag{6}$$

From $(\mathbf{I1})$, (4) and the first part of (5) we can start the following inference

$$\begin{aligned}
P &\sqsubseteq_{E_{\text{must}}} P' \sqcap \bigsqcap_{i\in I} P_i \\
&\sqsubseteq_{E_{\text{must}}} \bigsqcap_{i\in I} a_i.P_i' \quad \text{by } (\mathbf{Must2}) \\
&\sqsubseteq_{E_{\text{must}}} \bigsqcap_{i\in I} a_i.P_i'' \quad \text{by (5)} \\
&\sqsubseteq_{E_{\text{must}}} \bigsqcap_{i\in I} a_i.Q_i \quad \text{by (6)}
\end{aligned}$$

□