Real Reward Testing for Probabilistic Processes

Yuxin Deng[1] *, Rob van Glabbeek[2,4], Matthew Hennessy[3] † & Carroll Morgan[4] ‡

[1]Shanghai Jiao Tong University, China

[2]NICTA, Sydney, Australia

[3]Trinity College Dublin, Ireland

[4]University of New South Wales, Sydney, Australia

July 1, 2010

Abstract

We introduce a notion of real reward testing for probabilistic processes by extending the traditional nonnegative reward testing with negative rewards. In this testing framework, may and must preorders are the inverse relations of each other. More surprisingly, for finitary convergent processes, real reward must testing preorder coincides with nonnegative reward testing preorder. To prove the coincidence result, we characterise the usual resolution based testing in terms of weak derivations, and investigate some analytic properties such as the continuity of a function for calculating testing outcomes, and bounded continuity of a class of binary functions.

1 Introduction

Extending the classical testing semantics [6] to a setting with the co-existence of probability and nondeterminism was initiated by [15]. The application of a test to a process yields a set of probabilities for reaching some success states. In [14] a set Ω of possibly infinite number of success actions are used to report success. So the testing outcomes are vectors of probabilities of performing these success actions. In [5] we showed that for finitary processes, i.e. finite-state and finitely branching processes, vector based testing is no more powerful than scalar testing that employs only one success action. To prove that result we introduced a notion of reward testing inspired by [8]. The idea is to associate with each success action a nonnegative reward, and performing a success action means accumulating some reward. The outcomes of this reward testing are nonnegative expected rewards.

In certain occasions it is very natural to introduce negative rewards. For example, this is the case in the theory of Markove decision processes [12]. Intuitively, we could understand negative rewards as costs, while positive rewards are often viewed as benefits or profits. Then one would immediately pose the question: *if negative rewards are also allowed, how would the original reward testing semantics change?* We refer to the more relaxed form of testing as *real reward testing* and the original one as *nonnegative reward testing*. It has been shown in [5] that for finitary processes nonnegative reward may/must testing preorders ($\sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{may}}$ and $\sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}}$) coincide with purely probabilistic may/must testing preorders ($\sqsubseteq^{\Omega}_{\mathrm{pmay}}$ and $\sqsubseteq^{\Omega}_{\mathrm{pmust}}$); see Figure 1. In the present paper, we show that for real reward testing may and must preorders are the inverse of each other, i.e. for any processes P and Q,

$$P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{may}} Q \ \text{ iff } \ Q \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} P. \tag{1}$$

A more surprising result is that real reward must preorder coincides with nonnegative reward must preorder, i.e. for any finitary convergent processes P and Q,

$$P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q \ \text{ iff } \ P \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}} Q. \tag{2}$$

*Deng was supported by the National Natural Science Foundation of China (60703033).

†Hennessy gratefully acknowledges the financial support of Science Foundation Ireland (SFI 06 IN.1 1898).

‡Morgan acknowledges the support of ARC Discovery Grant DP0879529.

$$
\begin{array}{ccccccccccccc}
 & & & & \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{may}} & \stackrel{[5]}{=} & \sqsubseteq_{\mathrm{pmay}} & \stackrel{[5]}{=} & \sqsubseteq^{\Omega}_{\mathrm{pmay}} & \stackrel{[2]}{=} & \sqsubseteq_{S} \\
 & & & & \vdots & & & & & & \\
 & & & & \vdots & & & & & & \\
(\sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{may}})^{-1} & \stackrel{\mathit{Thm.}\ 3.8}{=} & \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} & \stackrel{\mathit{Thm.}\ 6.8}{=} & \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}} & \stackrel{[5]}{=} & \sqsubseteq_{\mathrm{pmust}} & \stackrel{[5]}{=} & \sqsubseteq^{\Omega}_{\mathrm{pmust}} & \stackrel{[2]}{=} & \sqsubseteq_{FS}
\end{array}
$$

The symbol $=$ between two relations means that they coincide, while a vertical dotted line between two relations denotes that the relation below is finer than the relation above if divergence is absent.

Figure 1: The relation of different testing preorders.

Here by convergence we mean that in the pLTS generated by a processes there is no infinite sequence of internal transitions between distributions like

$$\Delta_0 \xrightarrow{\tau} \Delta_1 \xrightarrow{\tau} \cdots$$

Although it is easy to see that in (2) the former is included in the latter, to prove that the latter is included in the former is far from being trivial. Our proof strategy is to characterise the usual resolution based testing approach as a kind of derivation based tesing, which allows us to exploit the failure simulation preorder $\sqsubseteq_{FS}$ that was proved in [2] to coincide with the purely probabilistic must testing preorder $\sqsubseteq^{\Omega}_{\mathrm{pmust}}$ based on resolutions, since it is relatively easy to demonstrate that $\sqsubseteq_{FS}$ is contained in $\sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}}$. Combining this with the results from [5] and [2], which show the coincidence of $\sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}}$ with $\sqsubseteq_{FS}$, leads to our required result that $\sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}}$ is included in $\sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}}$, as far as finitary convergent processes are concerned.

In order to establish the agreement of resolution based testing with derivation based testing, we need to show that in the first testing approach our results-collecting function in a deterministic labelled transition system is continuous, which in inturn requires us to identify a class of binary functions f with the property of *bounded continuity* in the sense that if the partial sum $\sum_{i=0}^{n} \lim_{j\to\infty} f(i,j)$, for any natural number n, is always bounded by a constant, then

$$\sum_{i=0}^{\infty} \lim_{j\to\infty} f(i,j) \;=\; \lim_{j\to\infty} \sum_{i=0}^{\infty} f(i,j),$$

that is, the limit and sum operations can be exchanged. We believe that this analytic result itself will be of independent interest.

Besides the above mentioned related work, a lot of other studies on probabilistic testing and simulation semantics have appeared in the literature. They are reviewed in [4, 3].

The rest of this paper is organised as follows. We start by recalling the syntax and operational semantics of a probabilistic extension of CSP from [2]. In Section 3 we review resolution based testing approach and show that real reward may preorder is simply the inverse relation of real reward must preorder. In Section 4 we present an alternative testing approach called derivation based testing. The two approaches are compared in Section 4. Then in Section 6 we show that for finitary processes real reward must testing coincides with nonnegative reward must testing. Finally, we conclude in Section 7.

2 The language pCSP

Let Act be a set of visible actions which a process can perform, and let Var be an infinite set of variables. The language pCSP of probabilistic CSP processes is given by the following two-sorted syntax, in which $p \in [0,1]$, $a \in \mathsf{Act}$ and $A \subseteq \mathsf{Act}$:

$$
\begin{array}{lcl}
P & ::= & S \mid P\,{}_{p}\!\oplus P \\
S & ::= & \mathbf{0} \mid x \in \mathsf{Var} \mid a.P \mid P \sqcap P \mid S \,\Box\, S \mid S \mid_A S \mid \mathsf{rec}\, x.\, P\,.
\end{array}
$$

This is the language introduced in [2]. The notions of free- and bound variables are standard; by $Q[x \mapsto P]$ we indicate substitution of term P for variable x in Q, with renaming if necessary. We write pCSP for the set of closed P-terms defined by this grammar, and sCSP for its *state-based* subset of closed S-terms.

The process $P \,_p\!\oplus Q$, for $0 \leq p \leq 1$, 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.[1] Any process is a probabilistic combination of state-based processes 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 $a.\_$ for $a \in \mathsf{Act}$, *internal*- and *external choices* $\sqcap$ and $\Box$, and a *parallel composition* $|_A$ for $A \subseteq \mathsf{Act}$.

The process $P \sqcap Q$ will first do a so-called *internal action* $\tau \notin \mathsf{Act}$, 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 same applies to $\mathsf{rec}\, x.\, P$ as, following CSP [11], our recursion construct performs an internal action when unfolding. As our testing semantics will abstract from internal actions, these τ-steps are harmless and merely simplify the semantics.

The process $s \Box t$ on the other hand does not perform actions itself but rather 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 requirement applies to $|_A$.

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

Although formally the operators $\Box$ and $|_A$ can only be applied to state-based processes, informally we use expressions of the form $P \Box Q$ and $P \;|_A\; Q$, where P and Q are *not* state-based, as syntactic sugar for expressions in the above syntax obtained by distributing $\Box$ and $|_A$ over $_p\!\oplus$. Thus for example $s \Box (t_1 \,_p\!\oplus t_2)$ abbreviates the term $(s \Box t_1) \,_p\!\oplus (s \Box t_2)$.

The full language of CSP [1, 7, 11] 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 $P \;|_A\; Q = (P \|_A Q)\backslash A$, where $\|_A$ and $\backslash A$ are the parallel composition and hiding operators of [11]. 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 all $p_i > 0$, and $\bigsqcap_{i\in I} P_i$, for some finite index set I.

Our language is interpreted as a *probabilistic labelled transition system* [4, 3]. Essentially the same model has appeared in the literature under different names such as *NP-systems* [8], *probabilistic processes* [9], *simple probabilistic automata* [13], *probabilistic transition systems* [10] etc. Furthermore, there are strong structural similarities with *Markov Decision Processes* [12, 5].

We now fix some notation. A (discrete) probability *subdistribution* over a set S is a function $\Delta : S \to [0,1]$ with $\sum_{s \in S} \Delta(s) \leq 1$; the *support* of such a Δ is $\lceil \Delta \rceil := \{\, s \in S \mid \Delta(s) > 0 \,\}$, and its *mass* $|\Delta|$ is $\sum_{s \in \lceil \Delta \rceil} \Delta(s)$. A subdistribution is a (total, or full) *distribution* if $|\Delta| = 1$. The point distribution $\overline{s}$ assigns probability 1 to s and 0 to all other elements of S, so that $\lceil \overline{s} \rceil = \{s\}$. With $\mathcal{D}(S)$ we denote the set of subdistributions over S, and with $\mathcal{D}_1(S)$ its subset of full distributions. For $\Delta, \Theta \in \mathcal{D}(S)$ we write $\Delta \leq \Theta$ iff $\Delta(s) \leq \Theta(s)$ for all $s \in S$.

Let $\{\Delta_k \mid k \in K\}$ be a set of subdistributions, possibly infinite. Then $\sum_{k\in K} \Delta_k$ is the real-valued function in $S \to \mathbb{R}$ defined by $(\sum_{k\in K} \Delta_k)(s) := \sum_{k\in K} \Delta_k(s)$. This is a partial operation on subdistributions because for some state s the sum of $\Delta_k(s)$ might exceed 1. If the index set is finite, say $\{1..n\}$, we often write $\Delta_1 + \ldots + \Delta_n$. For p a real number from $[0,1]$ we use $p{\cdot}\Delta$ to denote the subdistribution given by $(p{\cdot}\Delta)(s) := p{\cdot}\Delta(s)$. Finally we use ε to denote the everywhere-zero subdistribution that thus has empty support. These operations on subdistributions do not readily adapt themselves to distributions; yet if $\sum_{k\in K} p_k = 1$ for some collection of $p_k \geq 0$, and the Δ_k are distributions, then so is $\sum_{k\in K} p_k \cdot \Delta_k$. In general when $0{\leq}p{\leq}1$ we write $x \,_p\!\oplus y$ for $p{\cdot}x + (1{-}p){\cdot}y$ where that makes sense, so that for example $\Delta_1 \,_p\!\oplus \Delta_2$ is always defined, and is full if Δ_1 and Δ_2 are.

[1] In our semantics we have $[\![P \,_0\!\oplus Q]\!] = [\![Q]\!]$ and $[\![P \,_1\!\oplus Q]\!] = [\![P]\!]$, so without limitation of generality we could have required that $0{<}p{<}1$. In papers involving axiomatisations this is customary, as the most natural formulation of the law of associativity involves dividing by p.

(ACTION) $a.P \xrightarrow{a} [\![P]\!]$

(RECURSION) $\mathsf{rec}\, x.\, P \xrightarrow{\tau} [\![P[x \mapsto \mathsf{rec}\, x.\, P]]\!]$

(INT.L) $P \sqcap Q \xrightarrow{\tau} [\![P]\!]$

(INT.R) $P \sqcap Q \xrightarrow{\tau} [\![Q]\!]$

(EXT.L) $$\frac{s_1 \xrightarrow{a} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{a} \Delta}$$

(EXT.R) $$\frac{s_2 \xrightarrow{a} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{a} \Delta}$$

(EXT.I.L) $$\frac{s_1 \xrightarrow{\tau} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{\tau} \Delta \,\Box\, s_2}$$

(EXT.I.R) $$\frac{s_2 \xrightarrow{\tau} \Delta}{s_1 \,\Box\, s_2 \xrightarrow{\tau} s_1 \,\Box\, \Delta}$$

(PAR.L) $$\frac{s_1 \xrightarrow{\alpha} \Delta}{s_1 \,|_A\, s_2 \xrightarrow{\alpha} \Delta \,|_A\, s_2} \quad \alpha \notin A$$

(PAR.R) $$\frac{s_2 \xrightarrow{\alpha} \Delta}{s_1 \,|_A\, s_2 \xrightarrow{\alpha} s_1 \,|_A\, \Delta} \quad \alpha \notin A$$

(PAR.I) $$\frac{s_1 \xrightarrow{a} \Delta_1, \; s_2 \xrightarrow{a} \Delta_2}{s_1 \,|_A\, s_2 \xrightarrow{\tau} \Delta_1 \,|_A\, \Delta_2} \quad a \in A$$

In the above inferences A ranges over subsets of Act, and actions a, α are elements of $\mathsf{Act}, \mathsf{Act}_\tau$ respectively.

Figure 2: Operational semantics of pCSP

The expected value $\sum_{s\in S} \Delta(s) \cdot f(s)$ over a distribution Δ of a bounded non-negative function f to the reals or tuples of them is written $\mathrm{Exp}_\Delta(f)$, and the image of a distribution Δ through a function f is written $\mathrm{Img}_f(\Delta)$ — the latter is the distribution over the range of f given by $\mathrm{Img}_f(\Delta)(t) := \sum_{f(s)=t} \Delta(s)$.

Definition 2.1 A *probabilistic labelled transition system* (pLTS) is a triple $\langle S, L, \rightarrow\rangle$, where

(i) S is a set of states,
(ii) L is a set of transition labels,
(iii) relation $\rightarrow$ is a subset of $S \times L \times \mathcal{D}_1(S)$.

A (non-probabilistic) labelled transition system (LTS) may be viewed as a degenerate pLTS — one in which only point distributions are used. As with LTSs, we write $s \xrightarrow{\alpha} \Delta$ for $(s, \alpha, \Delta) \in \rightarrow$, as well as $s \xrightarrow{\alpha}$ for $\exists\Delta : s \xrightarrow{\alpha} \Delta$ and $s \rightarrow$ for $\exists\alpha : s \xrightarrow{\alpha}$. A pLTS is *deterministic* if for any state s and label α there is at most one distribution Δ with $s \xrightarrow{\alpha} \Delta$. A pLTS is *finitely branching* if the set $\{\langle\alpha, \Delta\rangle \mid s \xrightarrow{\alpha} \Delta,\ \alpha \in L\}$ is finite for all states s; if moreover S is finite, then the pLTS is *finitary*. A pLTS is *deterministic* if for each state s and label α, there is at most one distribution Δ with $s \xrightarrow{\alpha} \Delta$.

The operational semantics of pCSP is defined by a particular pLTS $\langle \mathsf{sCSP}, \mathsf{Act}_\tau, \rightarrow\rangle$ in which sCSP is the set of states and $\mathsf{Act}_\tau := \mathsf{Act} \cup \{\tau\}$ is the set of transition labels; we let a range over Act and α over Act_τ. We interpret pCSP processes P as distributions $[\![P]\!] \in \mathcal{D}_1(\mathsf{sCSP})$ via the function $[\![\_]\!] : \mathsf{pCSP} \rightarrow \mathcal{D}_1(\mathsf{sCSP})$ defined by

$$[\![s]\!] \;:=\; \overline{s} \quad \text{for } s \in \mathsf{sCSP}, \quad \text{and} \quad [\![P\,{}_p\!\oplus Q]\!] \;:=\; [\![P]\!]\,{}_p\!\oplus [\![Q]\!] \,.$$

The transition relation $\rightarrow$ is defined in Figure 2. This is a slight extension of the rules we used earlier [4, 3] for finite processes: one new rule is required to interpret recursive processes. All rules are very similar to the standard ones used to interpret CSP as a labelled transition system [11], but are modified so that the result of an action is a distribution. The rules for external choice and parallel composition use an obvious notation for distributing an operator over a distribution; for example $\Delta \,\Box\, s$ represents the distribution given by

$$(\Delta \,\Box\, s)(t) = \begin{cases} \Delta(s') & \text{if } t = s' \,\Box\, s \\ 0 & \text{otherwise.} \end{cases}$$

We sometimes write $\tau.P$ for $P \sqcap P$, thus giving $\tau.P \xrightarrow{\tau} [\![P]\!]$.

The set of states *reachable* from a subdistribution Δ is the smallest set that contains $\lceil\Delta\rceil$ and is closed under transitions, meaning that if some state s is reachable and $s \xrightarrow{\alpha} \Theta$ then every state in $\lceil\Theta\rceil$ is reachable as well. We graphically depict the operational semantics of a pCSP expression P by drawing the part of the pLTS reachable from $[\![P]\!]$ as a directed graph with states represented by filled nodes $\bullet$ and distributions by open nodes $\circ$. For any state s and distribution Δ with $s \xrightarrow{\alpha} \Delta$ we draw an edge from s to Δ labelled with α; and for any distribution Δ and state s in $\lceil\Delta\rceil$, the support of Δ, we draw an edge from Δ to s labelled with $\Delta(s)$. We often leave out point-distributions—diverting an incoming edge to the unique state in its support. Sometimes we partially unfold this graph by drawing the same nodes multiple times; in doing so, all outgoing edges of a given instance of a node are always drawn, but not necessarily all incoming edges.

Note that for each $P \in \mathsf{pCSP}$ the distribution $[\![P]\!]$ has finite support. Moreover, our pLTS is *finitely branching* in the sense that for each state $s \in \mathsf{sCSP}$ there are only finitely many pairs $(\alpha, \Delta) \in \mathsf{Act}_\tau \times \mathcal{D}_1(\mathsf{sCSP}))$ with $s \xrightarrow{\alpha} \Delta$. In spite of $[\![P]\!]$'s finite support, and the finite branching of our pLTS, it is possible for there to be infinitely many states reachable from $[\![P]\!]$; when there are only finitely many, then P is said to be finitary [5].

Definition 2.2 A subdistribution $\Delta \in \mathcal{D}(S)$ in a pLTS $\langle S, L, \rightarrow\rangle$ is *finitary* if only finitely many states are reachable from Δ; a pCSP expression P is *finitary* if $[\![P]\!]$ is. An expression P is *very finite* if $[\![P]\!]$ is finitary and it cannot reach any loop in the pLTS.

3 Testing probabilistic processes

We now retrace our earlier approach [4, 3] to the testing of probabilistic processes. A *test* is simply a very finite process in the language pCSP, except that it may in addition use special *success* actions for reporting outcomes: these are drawn from a set Ω of fresh actions not already in Act_τ. We refer to the augmented language as pCSP^Ω, and the pLTS it generates as $\mathsf{plts_t}$. Formally a test T is some very finite process from that language, and to apply test T to process P we form the process $T \;|_{\mathsf{Act}}\; P$ in which *all* visible actions of P must synchronise with T. The resulting composition is a process whose only possible actions are τ and the elements of Ω. We will define the result $\mathcal{A}(T, P)$ of applying the test T to the process P to be a set of testing outcomes, exactly one of which results from each resolution of the choices in $T \;|_{\mathsf{Act}}\; P$. Each *testing outcome* is an Ω-tuple of real numbers in the interval [0,1], i.e. a function $o : \Omega \rightarrow [0, 1]$, and its ω-component $o(\omega)$, for $\omega \in \Omega$, gives the probability that the resolution in question will reach an ω*-success state*, one in which the success action ω is possible.

Here we restrict attention to tests in which no state is simultaneously an ω-success state for different values of ω. In fact, we can go further by ruling out all tests in which from one success state one can reach another one, with a different success value.[2]

Definition 3.1 An Ω*-test* is a very finite closed pCSP expression T, but allowing the enriched alphabet $\mathsf{Act}_\tau \cup \Omega$ of actions instead of just Act_τ, such that if $t \xrightarrow{\omega_1}$ and $u \xrightarrow{\omega_2}$ for $\omega_1, \omega_2 \in \Omega$ with t reachable from T and u from t, then $\omega_1 = \omega_2$.

Due to the presence of nondeterminism in pLTSs, we need a mechanism to a reduce a nondeterministic structure into a set of deterministic structures, each of which determines a possible outcome. Here we adapt the notion of *resolution* defined in [14, 5] for probabilistic automata, to pLTSs.

Definition 3.2 [Resolutions] A *resolution* of a subdistribution $\Delta \in \mathcal{D}(S)$ in a pLTS $\langle S, \Omega_\tau, \rightarrow\rangle$ is a triple $\langle R, \Theta, \rightarrow_R\rangle$ where $\langle R, \Omega_\tau, \rightarrow_R\rangle$ is a deterministic pLTS and $\Theta \in \mathcal{D}(R)$, such that there exists a *resolving function* $f \in R \rightarrow S$ satisfying

(i) $\mathrm{Img}_f(\Theta) = \Delta$

(ii) if $r \xrightarrow{\alpha}_R \Theta'$ for $\alpha \in \Omega_\tau$ then $f(r) \xrightarrow{\alpha} \mathrm{Img}_f(\Theta')$

(iii) if $f(r) \xrightarrow{\alpha}$ for $\alpha \in \Omega_\tau$ then $r \xrightarrow{\alpha}_R$.

[2] Justification for imposing this restriction can be found in Appendix A of [5].

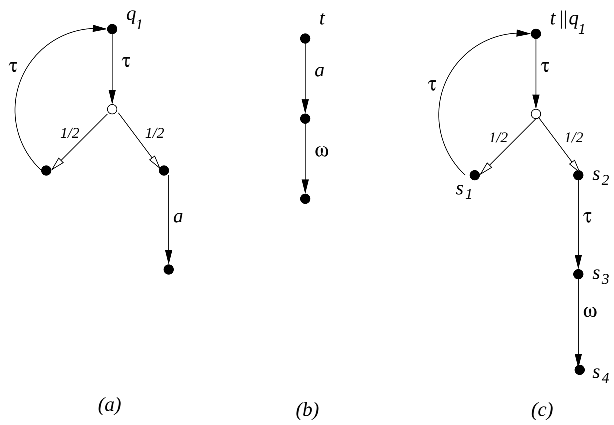

Figure 3: Testing the process Q_1

The reader is referred to Section 2 of [5] for a detailed discussion of this concept of resolutions, and the manner in which it represents *computation runs of a process*; in particular in a resolution states in S are allowed to be resolved into distributions, and computation steps can be *probabilistically interpolated*.

We now explain how to associate an outcome with a particular resolution, which in turn will associate a set of outcomes with a subdistribution in a pLTS. Given a deterministic pLTS $\langle R, \Omega_\tau, \rightarrow\rangle$ consider the functional $\mathcal{R} : (R \rightarrow [0,1]^\Omega) \rightarrow (R \rightarrow [0,1]^\Omega)$ defined by

$$\mathcal{R}(f)(r)(\omega) := \begin{cases} 1 & \text{if } r \xrightarrow{\omega} \\ 0 & \text{if } r \not\xrightarrow{\omega} \text{ and } r \not\xrightarrow{\tau} \\ \mathrm{Exp}_\Delta(f)(\omega) & \text{if } r \not\xrightarrow{\omega} \text{ and } r \xrightarrow{\tau} \Delta. \end{cases} \tag{3}$$

We view the unit interval $[0,1]$ ordered in the standard manner as a complete lattice; this induces the structure of a complete lattice on the product $[0,1]^\Omega$ and in turn on the set of functions $R \rightarrow [0,1]^\Omega$. The functional $\mathcal{R}$ is easily seen to be monotonic and therefore has a least fixed point, which we denote by $\mathbb{V}_{\langle R,\Omega_\tau,\rightarrow\rangle}$; this is abbreviated to $\mathbb{V}$ when the resolution in question is understood.

Now let $\mathcal{A}(T,P)$ denote the set of vectors

$$\{\, \mathrm{Exp}_\Theta(\mathbb{V}_{\langle R,\Omega_\tau,\rightarrow\rangle}) \mid \langle R, \Theta, \rightarrow\rangle \text{ is a resolution of } [\![T \mid_{\mathsf{Act}} P]\!] \,\}\ . \tag{4}$$

Example 3.3 Consider the process $Q_1 = \mathsf{rec}\, x.\, (\tau.x\ {}_{\frac{1}{2}}\!\oplus\ a.\mathbf{0})$ depicted in Figure 3(a). When we apply the test $T = a.\omega$ to it we get the pLTS in Figure 3(b), which is already deterministic, hence has only one resolution, itself. Moreover the outcome $\mathbb{V}$ associated with it is determined by its value at the state s_0. This in turn is the least solution of the equation

$$\mathbb{V}(s_0) = \frac{1}{2} \cdot \mathbb{V}(s_0) + \frac{1}{2}\vec{\omega}$$

In fact this equation has a unique solution in $[0,1]^\Omega$, namely $\vec{\omega}$, with $\vec{\omega}(\omega) = 1$ and $\vec{\omega}(\omega') = 0$ for all $\omega' \neq \omega$. Thus $\mathcal{A}(T, Q_1) = \{\vec{\omega}\}$.

□

Example 3.4 Consider the process $Q_2 = \mathsf{rec}\, x.\, (\tau.(x\ {}_{\frac{1}{2}}\!\oplus\ a) \ \Box\ \tau.(\mathbf{0}\ {}_{\frac{1}{2}}\!\oplus\ a))$ and the application of the test $T = a.\omega$ to it; this is outlined in Figure 4. In the pLTS of $T \mid_{\mathsf{Act}} Q_2$, for each $k \geq 1$ there is a resolution R_k such that $\mathbb{V}(R_k) = (1 - \frac{1}{2^k})\vec{\omega}$; intuitively it goes around the loop $(k-1)$ times before at last taking the right hand τ action. Thus $\mathcal{A}(T, Q_2)$ contains $(1 - \frac{1}{2^k})\vec{\omega}$ for every $k \geq 1$. But it also contains 1, because of the resolution which takes the left hand τ-move every time. Thus $\mathcal{A}(T, Q_2)$ includes the set

$$\{(1 - \frac{1}{2})\vec{\omega},\ (1 - \frac{1}{2^2})\vec{\omega}, \ldots, (1 - \frac{1}{2^k})\vec{\omega}, \ldots\ \vec{\omega}\}$$

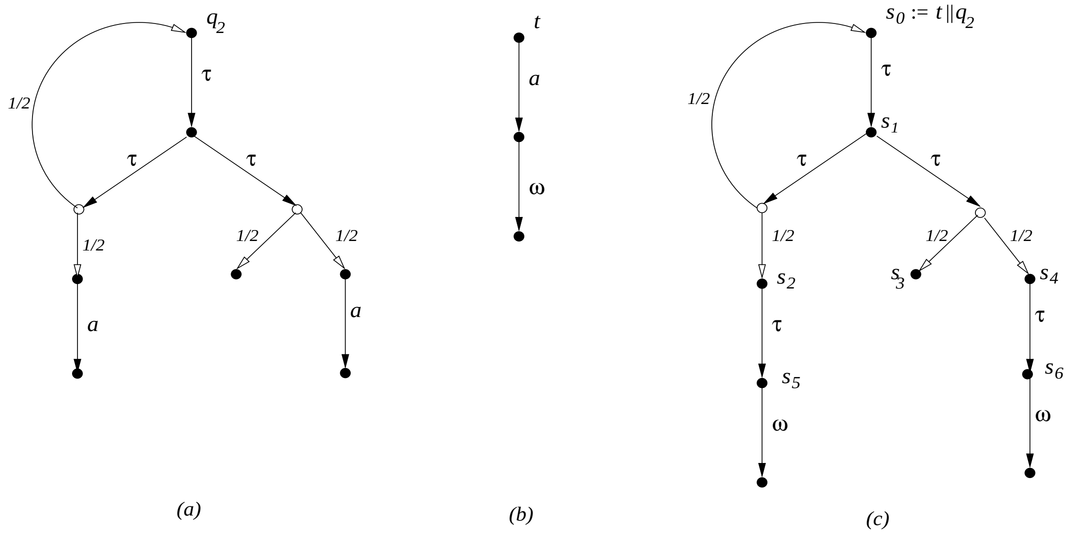

Figure 4: Testing the process Q_2

From later results it will follow that $\mathcal{A}(T, Q_2)$ is actually the convex closure of this set.

□

There are two standard methods for comparing two sets of outcomes:

$$O_1 \leq_{\mathrm{Ho}} O_2 \qquad \text{if for every } o_1 \in O_1 \text{ there exists some } o_2 \in O_2 \text{ such that } o_1 \leq o_2$$
$$O_1 \leq_{\mathrm{Sm}} O_2 \qquad \text{if for every } o_2 \in O_2 \text{ there exists some } o_1 \in O_1 \text{ such that } o_1 \leq o_2$$

This gives us our definition of the may- and must-testing preorders; they are decorated with $\cdot^{\Omega}$ for the repertoire Ω of testing actions they employ.

Definition 3.5 [Probabilistic testing preorders]

(i) $P \sqsubseteq^{\Omega}_{\text{pmay}} Q$ if for every Ω-test T, $\mathcal{A}(T, P) \leq_{\mathrm{Ho}} \mathcal{A}(T, Q)$.

(ii) $P \sqsubseteq^{\Omega}_{\text{pmust}} Q$ if for every Ω-test T, $\mathcal{A}(T, P) \leq_{\mathrm{Sm}} \mathcal{A}(T, Q)$.

These preorders are abbreviated to $P \sqsubseteq_{\text{pmay}} Q$, and $P \sqsubseteq_{\text{pmust}} Q$, when $|\Omega|= 1$.

In [5] we introduced two reward testing preorders. The idea is to associate each success action $\omega \in \Omega$ a reward, which is a nonnegative number in the unit interval $[0, 1]$, and a run of a probabilistic process in parallel with a test yields an expected reward accumulated by those states which can eanble success actions. A reward tuple $h \in [0, 1]^{\Omega}$ is used to assign reward $h(\omega)$ to success action ω, for each $\omega \in \Omega$. Due to the presence of nondeterminism, the application of a test T to a process P produces a set of expected rewards. Two sets of rewards can be compared by examing their supremum/infimum elements; this gives us two methods of testing called reward may/must testing. In [5] all rewards are required to be nonnegative, so we refer to that approach of testing as *nonnegative reward testing*. If we also allow negative rewards, which intuitively can be understood as costs, then we obtain an approach of testing called *real reward testing*. Technically, we simply let reward tuples h range over the set $[-1, 1]^{\Omega}$. If $o \in [0, 1]^{\Omega}$, we use the dot-product $h \cdot o = \sum_{\omega \in \Omega} h(\omega) * o(\omega)$. It can apply to a set $O \subseteq [0, 1]^{\Omega}$ so that $h \cdot A = \{h \cdot o \mid o \in O\}$. Let $A \subseteq [-1, 1]$. We use the notation $\bigsqcup A$ for the supremum of set A, and $\bigsqcap A$ for the infimum.

Definition 3.6 [Reward testing preorders]

(i) $P \sqsubseteq^{\Omega}_{\mathbf{nr}\text{may}} Q$ if for every Ω-test T and nonnegative reward tuple $h \in [0, 1]^{\Omega}$, $\bigsqcup h \cdot \mathcal{A}(T, P) \leq \bigsqcup h \cdot \mathcal{A}(T, Q)$.

(ii) $P \sqsubseteq^{\Omega}_{\mathbf{nr}\text{must}} Q$ if for every Ω-test T and nonnegative reward tuple $h \in [0, 1]^{\Omega}$, $\bigsqcap h \cdot \mathcal{A}(T, P) \leq \bigsqcap h \cdot \mathcal{A}(T, Q)$.

(iii) $P \sqsubseteq^{\Omega}_{\mathbf{rr}\text{may}} Q$ if for every Ω-test T and real reward tuple $h \in [-1,1]^{\Omega}$, $\bigsqcup h \cdot \mathcal{A}(T,P) \leq \bigsqcup h \cdot \mathcal{A}(T,Q)$.

(iv) $P \sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}} Q$ if for every Ω-test T and real reward tuple $h \in [-1,1]^{\Omega}$, $\bigsqcap h \cdot \mathcal{A}(T,P) \leq \bigsqcap h \cdot \mathcal{A}(T,Q)$.

It is shown in Corollary 1 of [5] that nonnegative reward testing is equally powerful as probabilistic testing.

Theorem 3.7 [5] For any finitary processes P and Q,

(i) $P \sqsubseteq^{\Omega}_{\mathbf{nr}\text{may}} Q$ if and only if $P \sqsubseteq^{\Omega}_{\text{pmay}} Q$.

(ii) $P \sqsubseteq^{\Omega}_{\mathbf{nr}\text{must}} Q$ if and only if $P \sqsubseteq^{\Omega}_{\text{pmust}} Q$. □

In this paper we focus on the real reward testing preorders $\sqsubseteq^{\Omega}_{\mathbf{rr}\text{may}}$ and $\sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}}$, by comparing them with the nonnegative reward testing preorders $\sqsubseteq^{\Omega}_{\mathbf{nr}\text{may}}$ and $\sqsubseteq^{\Omega}_{\mathbf{nr}\text{must}}$. We first show that, although the two nonnegative reward testing preorders are in general incomparable, the two real reward testing preorders are simply the inverse relations of each other.

Theorem 3.8 For any processes P and Q, it holds that $P \sqsubseteq^{\Omega}_{\mathbf{rr}\text{may}} Q$ if and only if $Q \sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}} P$.

Proof: We first notice that for any nonempty set $A \subseteq [0,1]^{\Omega}$ and any reward tuple $h \in [-1,1]^{\Omega}$,

$$\bigsqcup h \cdot A \quad = \quad -(\bigsqcap (-h) \cdot A) \tag{5}$$

where $-h$ is the negation of h, i.e. $(-h)(\omega) = -(h(\omega))$ for any $\omega \in \Omega$. We consider the "if" direction; the "only if" direction is similar. Let T be any Ω-test and h be any real reward tuple in $[-1,1]^{\Omega}$. Clearly, $-h$ is also a real reward tuple. Suppose $Q \sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}} P$, then

$$\bigsqcap (-h) \cdot \mathcal{A}(T,Q) \quad \leq \quad \bigsqcap (-h) \cdot \mathcal{A}(T,P) \tag{6}$$

Therefore, we can infer that

$$\begin{array}{rcll} \bigsqcup h \cdot \mathcal{A}(T,P) & = & -(\bigsqcap (-h) \cdot \mathcal{A}(T,P)) & \text{by (5)} \\ & \leq & -(\bigsqcap (-h) \cdot \mathcal{A}(T,Q)) & \text{by (6)} \\ & = & \bigsqcup h \cdot \mathcal{A}(T,Q) & \text{by (5)} \end{array}$$

□

Our next task is to compare $\sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}}$ with $\sqsubseteq^{\Omega}_{\mathbf{nr}\text{must}}$. The former is included in the latter, which directly follows from Definition 3.6. Surprisingly, it turns out that for finitary convergent processes the latter is also included in the former, thus the two preorders are in fact the same. The rest of the paper is devoted to proving this result.

4 Derivation based testing

In this section we give an alternative definition of $\mathcal{A}(T,P)$. Our definition has four ingredients. First of all, for technical reasons we normalise our pLTS by removing all τ-transitions that leave a success state. This way an ω-success state will only have outgoing transitions labelled ω.

Definition 4.1 [ω-respecting]. Let $\langle S, L, \rightarrow \rangle$ be a pLTS such that the set of labels L includes Ω. It is said to be *ω-respecting* whenever $s \xrightarrow{\omega}$, for any $\omega \in \Omega$, $s \stackrel{\tau}{\not\rightarrow}$.

It is straightforward to modify an arbitrary pLTS so that it is ω-respecting. Here we outline how this is done for our pLTS for pCSP.

Definition 4.2 [Pruning] Let $[\cdot]$ be the unary operator on Ω-test states given by the operational rules

$$\frac{s \xrightarrow{\omega} \Delta}{[s] \xrightarrow{\omega} [\Delta]}\ (\omega \in \Omega) \qquad\qquad \frac{s \stackrel{\omega}{\not\rightarrow}\ (\text{for all } \omega \in \Omega),\ \ s \xrightarrow{\alpha} \Delta}{[s] \xrightarrow{\alpha} [\Delta]}\ (\alpha \in \mathsf{Act}_{\tau})\ .$$

Just as $\Box$ and $|_A$, this operator extends as syntactic sugar to Ω-tests by distributing $[\cdot]$ over ${}_p\oplus$; likewise, it extends to distributions by $[\Delta]([s]) = \Delta(s)$. Clearly, this operator does nothing else than removing all outgoing transitions of a success state other than the ones labelled with $\omega \in \Omega$.

Secondly, we recall the definition of weak derivations from [2]. In a pLTS actions are only performed by states, in that actions are given by relations from states to distributions. But pCSP processes in general correspond to distributions over states, so in order to define what it means for a process to perform an action, we need to *lift* these relations so that they also apply to distributions. In fact we will find it convenient to lift them to subdistributions.

Definition 4.3 Let $(S, L, \rightarrow)$ be a pLTS and $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ be a relation from states to subdistributions. Then $\overline{\mathcal{R}} \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$ is the smallest relation that satisfies:

(i) $s \; \mathcal{R} \; \Theta$ implies $\overline{s} \; \overline{\mathcal{R}} \; \Theta$, and

(ii) (Linearity) $\Delta_i \; \overline{\mathcal{R}} \; \Theta_i$ for $i \in I$ implies $(\sum_{i\in I} p_i \cdot \Delta_i) \; \overline{\mathcal{R}} \; (\sum_{i\in I} p_i \cdot \Theta_i)$ for any $p_i \in [0,1]$ $(i \in I)$ with $\sum_{i\in I} p_i \leq 1$.

An application of this notion is when the relation is $\xrightarrow{\alpha}$ for $\alpha \in \mathsf{Act}_\tau$; in that case we also write $\xrightarrow{\alpha}$ for $\overline{\xrightarrow{\alpha}}$. Thus, as source of a relation $\xrightarrow{\alpha}$ we now also allow distributions, and even subdistributions. A subtlety of this approach is that for any action α, we have

$$\varepsilon \xrightarrow{\alpha} \varepsilon \tag{7}$$

simply by taking $I = \emptyset$ or $\sum_{i\in I} p_i = 0$ in Definition 4.3. That will turn out to make ε especially useful for modelling the "chaotic" aspects of divergence, in particular that in the must-case a divergent process can simulate any other.

Definition 4.4 [Weak derivatives] Suppose we have subdistributions $\Delta,\ \Delta_k,\ \Delta_k^{\rightarrow}, \Delta_k^{\times}$, for $k \geq 0$, with the following properties:

$$\begin{array}{lcll}
\Delta & = & \Delta_0 = \Delta_0^{\rightarrow} + \Delta_0^{\times} & \text{— The } \times \text{ component stops "here" (even if it could have moved),} \\
\Delta_0^{\rightarrow} & \xrightarrow{\tau} & \Delta_1 = \Delta_1^{\rightarrow} + \Delta_1^{\times} & \text{— but the } \rightarrow \text{ component moves on.} \\
\vdots & & \qquad\qquad\quad \vdots & \\
\Delta_k^{\rightarrow} & \xrightarrow{\tau} & \Delta_{k+1} = \Delta_{k+1}^{\rightarrow} + \Delta_{k+1}^{\times} & \\
 & & \qquad\qquad\quad \vdots & \\
\text{In total:} & & \Delta' = \sum_{k=0}^{\infty} \Delta_k^{\times} & \text{— Finally, all the stopped-somewhere components are summed.}
\end{array}$$

The $\xrightarrow{\tau}$ moves above with subdistribution sources are lifted in the sense of the previous section.

We call $\Delta' := \sum_{k=0}^{\infty} \Delta_k^{\times}$ a *weak derivative* of Δ, and write $\Delta \Longrightarrow \Delta'$ to mean that Δ can make a *weak τ move* to its derivative Δ'.

There is always at least one derivative of any distribution (the distribution itself) and there can be many.

Thirdly, we identify a class of special weak derivatives called extreme derivatives.

Definition 4.5 [Extreme derivatives] A state s in a pLTS is called *stable* if $s \stackrel{\tau}{\not\longrightarrow}$, and a subdistribution Θ is called *stable* if every state in its support is stable. We write $\Delta \Longrightarrow\!\!\!\succ \Theta$ whenever $\Delta \Longrightarrow \Theta$ and Θ is stable, and call Θ an *extreme* derivative of Δ.

Referring to Definition 4.4, we see this means that in the extreme derivation of Θ from Δ at every stage a state must move on if it can, so that every stopping component can contain only states which *must* stop: for $s \in \Delta_k^{\rightarrow} + \Delta_k^{\times}$ we have $s \in \Delta_k^{\times}$ if *and now also* only if $s \stackrel{\tau}{\not\longrightarrow}$. Moreover if the pLTS is ω-respecting then whenever $s \in \Delta_k^{\rightarrow}$, that is whenever it marches on, it is not successful, $s \stackrel{\omega}{\not\longrightarrow}$ for every $\omega \in \Omega$.

Lemma 4.6 [Existence of extreme derivatives]

(i) For every subdistribution Δ there exists some (stable) Δ' such that $\Delta \Longrightarrow\!\!\!\succ \Delta'$.

(ii) In a deterministic pLTS we have that $\Delta \Longrightarrow\!\!\!\succ \Delta'$ and $\Delta \Longrightarrow\!\!\!\succ \Delta''$ implies $\Delta' = \Delta''$.

Proof: We construct a derivation as in Definition 4.4 of a stable Δ' by defining the components $\Delta_k, \Delta_k^\times$ and $\Delta_k^\rightarrow$ using induction on k. Let us assume that the subdistribution Δ_k has been defined; in the base case $k = 0$ this is simply Δ. The decomposition of this Δ_k into the components $\Delta_k^\times$ and $\Delta_k^\rightarrow$ is carried out by defining the former to be precisely those states which must stop, i.e. those s for which $s \stackrel{\tau}{\not\longrightarrow}$. Formally $\Delta_k^\times$ is determined by:

$$\Delta_k^\times(s) = \begin{cases} \Delta_k(s) & \text{if } s \stackrel{\tau}{\not\longrightarrow} \\ 0 & \text{otherwise} \end{cases}$$

Then $\Delta_k^\rightarrow$ is given by the *remainder* of Δ_k, namely those states which can perform a τ action:

$$\Delta_k^\rightarrow(s) = \begin{cases} \Delta_k(s) & \text{if } s \stackrel{\tau}{\longrightarrow} \\ 0 & \text{otherwise} \end{cases}$$

Note that these definitions divide the support of Δ_k into two disjoints sets, namely the support of $\Delta_k^\times$ and the support of $\Delta_k^\rightarrow$. Moreover by construction we know that $\Delta_k^\rightarrow \stackrel{\tau}{\longrightarrow} \Theta$ for some Θ; we let Δ_{k+1} be an arbitrary such Θ.

This completes our definition of an extreme derivative as in Definition 4.4 and so we have established (i).

For (ii) we observe that in a deterministic pLTS the above choice of Δ_{k+1} is unique, so that the whole derivative construction becomes unique. □

It is worth pointing out that the use of subdistributions, rather than distributions, is essential here. Consider a state t that has only one transition, a self τ-loop $t \stackrel{\tau}{\longrightarrow} \overline{t}$. Then it diverges and it has a unique extreme derivative ε, the empty subdistribution. More generally, suppose a subdistribution Δ diverges, that is there is an infinite sequence of derivations $\Delta \stackrel{\tau}{\longrightarrow} \Delta_1 \stackrel{\tau}{\longrightarrow} \ldots \Delta_k \stackrel{\tau}{\longrightarrow} \ldots$. Then one extreme derivative of Δ is ε, but it may have others.

The final ingredient in the definition of a set of outcomes is to use this notion of extreme derivative to formalise the subdistributions that can be reached from $[\![T \mid_{\mathsf{Act}} P]\!]$. Note that all states $s \in \lceil\Theta\rceil$ in the support of an extreme derivative either satisfy $s \stackrel{\omega}{\longrightarrow}$ for a unique $\omega \in \Omega$, or have $s \not\longrightarrow$.

Definition 4.7 [Outcomes] The outcome $\$\Theta \in [0,1]^\Omega$ of a stable subdistribution Θ is given by $\$\Theta(\omega) = \sum_{s \in \lceil\Theta\rceil,\ s \stackrel{\omega}{\longrightarrow}} \Theta(s)$.

Putting all four ingredients together, we arrive at a definition of $\mathcal{A}^{\mathrm{d}}(T, P)$:

Definition 4.8 Let P be a pCSP process and T an Ω-test. Then $\mathcal{A}^{\mathrm{d}}(T, P) = \{\$\Theta \mid [\![T \mid_{\mathsf{Act}} P]\!] \Longrightarrow\!\!\!\!\succ \Theta\}$.

The role of pruning in the above definition can be seen via the following example.

Example 4.9 Let $P = a.b.\mathbf{0}$ and $T = a.(b.\mathbf{0} \,\Box\, \omega.\mathbf{0})$. The pLTS generated by applying T to P can be described by the process $\tau.(\tau.\mathbf{0} \,\Box\, \omega.\mathbf{0})$. Then $[\![T \mid_{\mathsf{Act}} P]\!]$ has a unique extreme derivation $[\![T \mid_{\mathsf{Act}} P]\!] \Longrightarrow\!\!\!\!\succ [\![\mathbf{0}]\!]$, and $[[\![T \mid_{\mathsf{Act}} P]\!]]$ also has a unique extreme derivation $[[\![T \mid_{\mathsf{Act}} P]\!]] \Longrightarrow\!\!\!\!\succ [\![\omega.\mathbf{0}]\!]$. The outcome in $\mathcal{A}^{\mathrm{d}}(T, P)$ shows that process P passes test T with probability 1, which is what we expect for state-based testing, which we use in this paper. Without pruning we would get an outcome saying that P passes T with probability 0. □

Example 4.10 (Revisiting Example 3.3.) The pLTS in Figure 3(b) is deterministic and unaffected by pruning; from part (ii) of Lemma 4.6 it follows that $\overline{s_0}$ has a unique extreme derivative Θ. Moreover Θ can be calculated to be

$$\sum_{k \geq 1} \frac{1}{2^k} \cdot \overline{\omega},$$

which simplifies to the distribution $\overline{\omega}$. Thus is the same set of results gained by applying T to $a.\mathbf{0}$ on its own. □

Example 4.11 (Revisiting Example 3.4.) The application of the test T to processes Q_2 is outlined in Figure 4(b). Consider any extreme derivative Δ' from $[[\![T \mid_{\mathsf{Act}} Q_2]\!]]$, which we have abbreviated to $\overline{s_0}$; note that here again pruning actually has no effect. Using the notation of Definition 4.4, it is clear that $\Delta_0^\times$ and $\Delta_0^\rightarrow$ must be ε and $\overline{s_0}$ respectively. Similarly, $\Delta_1^\times$ and $\Delta_1^\rightarrow$ must be ε and $\overline{s_1}$ respectively. But s_1 is a nondeterministic state, having two possible transitions:

(i) $s_1 \xrightarrow{\tau} \Theta_0$ where Θ_0 has support $\{s_0, s_2\}$ and assigns each of them the weight $\frac{1}{2}$

(ii) $s_1 \xrightarrow{\tau} \Theta_1$ where Θ_1 has the support $\{s_3, s_4\}$, again diving the mass equally among them.

So there are many possibilities for Δ_2; it is easy to see from Definition 4.4 that in fact Δ_2 can be of the form

$$p \cdot \Theta_0 + (1-p) \cdot \Theta_1 \tag{8}$$

for any choice of $p \in [0,1]$.

Let us consider one possibility, an extreme one where p is chosen to be 0; only the transition (ii) above is used. Here $\Delta_2^{\rightarrow}$ is the subdistribution $\frac{1}{2}\overline{s_4}$, and $\Delta_k^{\rightarrow} = \varepsilon$ whenever $k > 2$. A simple calculation shows that in this case the extreme derivative generated is $\Theta_1^e = \frac{1}{2}\overline{s_3} + \frac{1}{2}\overline{\omega}$ which implies that $\frac{1}{2}\vec{\omega} \in \mathcal{A}^{\mathrm{d}}(T, Q_2)$.

Another possibility for Δ_2 is Θ_0, corresponding to the choice of $p = 1$ in (8) above. Continuing with this derivation leads to Δ_3 being $\frac{1}{2} \cdot \overline{s_1} + \frac{1}{2} \cdot \overline{\omega}$; in other words $\Delta_3^{\times} = \frac{1}{2} \cdot \overline{\omega}$ and $\Delta_3^{\rightarrow} = \frac{1}{2} \cdot \overline{s_1}$. Now in the generation of Δ_4 from $\Delta_3^{\rightarrow}$ once more we have to resolve a transition from the nondeterministic state s_1, by choosing some arbitrary $p \in [0,1]$ in (8). Suppose that each time this arises we systematically choose $p = 1$, that is, we ignore completely the transition (ii) above. Then it is easy to see that the extreme derivative generated is

$$\Theta_0^e = \sum_{k \geq 1} \frac{1}{2^k} \cdot \overline{\omega}$$

which simplifies to the distribution $\overline{\omega}$. This in turn means that $\vec{\omega} \in \mathcal{A}^{\mathrm{d}}(T, Q_2)$.

We have seen two possible derivations of extreme derivatives from $\overline{s_0}$. But there are many others. In general whenever $\Delta_k^{\rightarrow}$ is of the form $q \cdot \overline{s_1}$ we have to resolve the nondeterminism by choosing a $p \in [0,1]$ in (8) above; moreover each such choice is independent. However it follows from Theorem 6 in [2] that every extreme derivative Δ' of $\overline{s_0}$ is of the form

$$q \cdot \Theta_0^e + (1-q)\Theta_1^e$$

for some choice of $q \in [0,1]$. Consequently it follows that $\mathcal{A}^{\mathrm{d}}(T, Q_2)$ is the convex closure of the set $\{\frac{1}{2}\vec{\omega}, \vec{\omega}\}$.

□

5 Comparison of resolution and derivation based testing

We have now seen two ways of associating sets of outcomes with the application of a test to the process. The first, in Section 3, we associate with a test and a process a set of deterministic structures called resolutions, while in the second, in Section 4, uses extreme derivations in which nondeterministic choices are resolved dynamically as the derivation proceeds. In this section we show that the testing preorders obtained from these two approaches coincide.

First let us see how an extreme derivation can be viewed as a method for dynamically generating a resolution.

Theorem 5.1 [Resolutions from extreme derivatives] Suppose $\Delta \Longrightarrow\!\!\!\succ \Delta'$ in a pLTS $\langle S, \Omega_\tau, \rightarrow \rangle$. Then there is a resolution $\langle R, \Theta, \rightarrow_R \rangle$ of Δ, with resolving function f, such that $\Theta \Longrightarrow\!\!\!\succ_R \Theta'$ for some Θ' such that $\Delta' = \mathrm{Img}_f(\Theta')$.

Proof: Consider an extreme derivation of $\Delta \Longrightarrow\!\!\!\succ \Delta'$, as given in Definition 4.4 where all $\Delta_k^{\times}$ are assumed to be stable. To define the corresponding resolution $\langle R, \Theta, \longrightarrow_R \rangle$ we refer to Definition 3.2. First let the set of states R be $S \times \mathbb{N}$ and the resolving function $f \colon R \to S$ be given by $f(s,k) = s$. To complete the description we must define the partial functions $\xrightarrow{\alpha}$, for $\alpha = \omega$ and $\alpha = \tau$. These are always defined so that if $(s,k) \xrightarrow{\alpha} \Gamma$ then the only states in the support of Γ are of the form $(s', k+1)$. In the definition we use $\Theta^{\downarrow k}$, for any subdistribution Θ over S, to be the subdistribution over R given by

$$\Theta^{\downarrow k}(t) = \begin{cases} \Theta(s) & \text{if } t = (s,k) \\ 0 & \text{otherwise} \end{cases}$$

Note that by definition

(a) $\mathrm{Img}_f(\Theta^{\downarrow k}) = \Theta$

(b) $\Delta_k^{\downarrow k} = \Delta_k^{\rightarrow\downarrow k} + \Delta_k^{\times\downarrow k}$

The definition of $\xrightarrow{\omega}_R$ is straightforward: its domain consists of states (s,k) where $s \in \lceil \Delta_k^\times \rceil$ and is defined by letting $(s,k) \xrightarrow{\omega} \Delta_s^{\downarrow k+1}$ for some arbitrarily chosen $s \xrightarrow{\omega} \Delta_s$.

The definition of $\xrightarrow{\tau}_R$ is more complicated, and is determined by the moves $\Delta_k^\rightarrow \xrightarrow{\tau} \Delta_{k+1}$. For a given k this move means that

$$\Delta_k^\rightarrow = \sum_{i\in I} p_i \cdot \overline{s_i}, \qquad \Delta_{k+1} = \sum_{i\in I} p_i \cdot \Gamma_i, \qquad s_i \xrightarrow{\tau} \Gamma_i$$

So for each k we let

$$(s,k) \xrightarrow{\tau}_R \sum_{s_i = s} p_i \cdot \Gamma_i^{\downarrow k+1}$$

This definition ensures

(c) $(\Delta_k^\rightarrow)^{\downarrow k} \xrightarrow{\tau}_R (\Delta_{k+1})^{\downarrow k+1}$

(d) $(\Delta_k^\times)^{\downarrow k}$ is stable.

This completes our definition of the deterministic pLTS underlying the required resolution; it remains to find distributions Θ, Θ' over R such that $\Delta = \mathrm{Img}_f(\Theta)$, $\Delta' = \mathrm{Img}_f(\Theta')$ and $\Theta \Longrightarrow\!\!\!\succ \Theta'$.

Because of (b) (c) and (d) we have the following extreme derivation, which by part (ii) of Lemma 4.6 is the unique one from $\Delta_0^{\downarrow 0}$:

$$\begin{array}{lcrcl}
\Delta^{\downarrow 0} & = & (\Delta_0^\rightarrow)^{\downarrow 0} & + & (\Delta_0^\times)^{\downarrow 0} \\
(\Delta_0^\rightarrow)^{\downarrow 0} & \xrightarrow{\tau}_R & (\Delta_1^\rightarrow)^{\downarrow 1} & + & (\Delta_1^\times)^{\downarrow 1} \\
\vdots & & & & \vdots \\
(\Delta_k^\rightarrow)^{\downarrow k} & \xrightarrow{\tau}_R & (\Delta_{k+1}^\rightarrow)^{\downarrow k+1} & + & (\Delta_{k+1}^\times)^{\downarrow k+1} \\
 & & & & \vdots \\
\hline
 & & & & \Theta' = \sum_{k=0}^\infty (\Delta_k^\times)^{\downarrow k}
\end{array}$$

Letting Θ be $\Delta^{\downarrow 0}$, we see that (a) above ensures $\Delta = \mathrm{Img}_f(\Theta)$; the same note and the linearity of f applied to distributions also gives $\Delta' = \mathrm{Img}_f(\Theta')$.

□

The converse is somewhat simpler.

Proposition 5.2 [Extreme derivatives from resolutions] Suppose $\langle R, \Theta, \rightarrow_R\rangle$ is a resolution of a subdistribution Δ in a pLTS $\langle S, \Omega_\tau, \rightarrow\rangle$ with resolving function f. Then $\Theta \Longrightarrow\!\!\!\succ_R \Theta'$ implies $\Delta \Longrightarrow\!\!\!\succ \mathrm{Img}_f(\Theta')$.

Proof: Consider any derivation of $\Theta \Longrightarrow\!\!\!\succ_R \Theta'$ along the lines of Definition 4.4. By systematically applying the function f to the component subdistributions in this derivation we get a derivation $\mathrm{Img}_f(\Theta) \Longrightarrow \mathrm{Img}_f(\Theta')$, that is $\Delta \Longrightarrow \mathrm{Img}_f(\Theta')$. To show that $\mathrm{Img}_f(\Theta')$ is actually an extreme derivative it suffices to show that s is stable for every $s \in \lceil \mathrm{Img}_f(\Theta') \rceil$. But if $s \in \lceil \mathrm{Img}_f(\Theta') \rceil$ then by definition there is some $t \in \lceil \Theta' \rceil$ such that $s = f(t)$. Since $\Theta \Longrightarrow\!\!\!\succ_R \Theta'$ the state t must be stable. The stability of s now follows from requirement (iii) of Definition 3.2. □

Our next step is to relate the outcomes extracted from extreme derivatives to those extracted from the corresponding resolutions. This requires some analysis of the evaluation function $\mathbb{V}(-)$. We show that the function $\mathcal{R}$ defined in (3) and its least fixed point $\mathbb{V}(-)$ are continuous.

Definition 5.3 [Continuous functions] A chain in a complete lattice L is a sequence of elements $\{ c_n \mid n \geq 0 \}$ satisfying $c_i \leq c_{i+1}$. Obviously chains have least upper bounds which we denote by $\bigsqcup_{n \geq 0} c_n$. A function $f : L \to L$ is said to be continuous if it preserves the least upper bounds of chains

$$f(\bigsqcup_{n \geq 0} c_n) = \bigsqcup_{n \geq 0} f(c_n)$$

The following technical lemma states that some real functions satisfy the property of *bounded continuity*, which allows the exchange of limit and sum operations. It plays a crucial role in proving the continuity of $\mathcal{R}$.

Proposition 5.4 [Bounded continuity] Given a function $f : \mathbb{N} \times \mathbb{N} \to \mathbb{R}_{\geq 0}$ which satisfies the following conditions

C1. f is monotonic on the second parameter, i.e. $j_1 \leq j_2$ implies $f(i, j_1) \leq f(i, j_2)$ for all $i, j_1, j_2 \in \mathbb{N}$.

C2. For any $i \in \mathbb{N}$, the limit $\lim_{j \to \infty} f(i, j)$ exists.

C3. For any $n \in \mathbb{N}$, the partial sum $S_n = \sum_{i=0}^{n} \lim_{j \to \infty} f(i, j)$ is bounded, i.e. there exists some $c \in \mathbb{R}_{\geq 0}$ such that $S_n \leq c$ for all $n \geq 0$.

then it holds that

$$\sum_{i=0}^{\infty} \lim_{j \to \infty} f(i, j) \; = \; \lim_{j \to \infty} \sum_{i=0}^{\infty} f(i, j).$$

Proof: Let ϵ be any positive real number. By **C3** the sequence $\{S_n\}_{n=0}^{\infty}$ is bounded and it is nondecreasing, so it converges to $\sum_{i=0}^{\infty} \lim_{j \to \infty} f(i, j)$. Then there exists some $n_\epsilon \in \mathbb{N}$ such that

$$0 \; \leq \; \sum_{i=0}^{\infty} \lim_{j \to \infty} f(i, j) - \sum_{i=0}^{n_\epsilon} \lim_{j \to \infty} f(i, j) \; \leq \; \frac{\epsilon}{2}. \tag{9}$$

By **C1** and **C2**, for any $i \in \mathbb{N}$, the sequence $\{f(i, j)\}_{j=0}^{\infty}$ is nondecreasing and converges to $\lim_{j \to \infty} f(i, j)$. Therefore, for each $i \in \mathbb{N}$, there exists some $m_{i,\epsilon,n_\epsilon} \in \mathbb{N}$ such that

$$\forall j \geq m_{i,\epsilon,n_\epsilon} : \qquad 0 \; \leq \; \lim_{j \to \infty} f(i, j) - f(i, j) \; \leq \; \frac{\epsilon}{2n_\epsilon}. \tag{10}$$

Let $m_\epsilon = \max\{m_{i,\epsilon,n_\epsilon} \mid 0 \leq i \leq n_\epsilon\}$. It follows from (10) that

$$\forall j \geq m_\epsilon : \qquad 0 \; \leq \; \sum_{i=0}^{n_\epsilon} \lim_{j \to \infty} f(i, j) - \sum_{i=0}^{n_\epsilon} f(i, j) \; \leq \; \frac{\epsilon}{2}. \tag{11}$$

By summing up (9) and (11), we obtain

$$\forall j \geq m_\epsilon : \qquad 0 \; \leq \; \sum_{i=0}^{\infty} \lim_{j \to \infty} f(i, j) - \sum_{i=0}^{n_\epsilon} f(i, j) \; \leq \; \epsilon. \tag{12}$$

By **C1** and **C2**, we have that $f(i, j) \leq \lim_{j \to \infty} f(i, j)$ for any $i, j \in \mathbb{N}$. So for any $j, n \in \mathbb{N}$ the partial sum $\sum_{i=0}^{n} f(i, j)$ is bounded as

$$\sum_{i=0}^{n} f(i, j) \; \leq \; \sum_{i=0}^{n} \lim_{j \to \infty} f(i, j) \; \leq \; c$$

according to **C3**. Thus it converges to $\sum_{i=0}^{\infty} f(i, j)$. Then for any $j \in \mathbb{N}$ there exists some $n_{j,\epsilon}$ such that

$$\forall n \geq n_{j,\epsilon} : \qquad 0 \; \leq \; \sum_{i=0}^{\infty} f(i, j) - \sum_{i=0}^{n} f(i, j) \; \leq \; \epsilon. \tag{13}$$

Now consider the particular case that $j = m_\epsilon$. Let $N_\epsilon = \max\{n_\epsilon, n_{m_\epsilon,\epsilon}\}$. We know from (12) that

$$0 \;\leq\; \sum_{i=0}^{\infty} \lim_{j\to\infty} f(i,j) - \sum_{i=0}^{N_\epsilon} f(i, m_\epsilon) \;\leq\; \epsilon. \tag{14}$$

From (13) we infer that

$$-\epsilon \;\leq\; \sum_{i=0}^{N_\epsilon} f(i, m_\epsilon) - \sum_{i=0}^{\infty} f(i, m_\epsilon) \;\leq\; 0. \tag{15}$$

By summing up (14) and (15), we derive that

$$-\epsilon \;\leq\; \sum_{i=0}^{\infty} \lim_{j\to\infty} f(i,j) - \sum_{i=0}^{\infty} f(i, m_\epsilon) \;\leq\; \epsilon. \tag{16}$$

We conclude from (16) that

$$\lim_{j\to\infty} \sum_{i=0}^{\infty} f(i,j) \;=\; \sum_{i=0}^{\infty} \lim_{j\to\infty} f(i,j).$$

□

Lemma 5.5 Consider a deterministic pLTS $\langle R, \Omega_\tau, \rightarrow\rangle$. The function $\mathcal{R}$ defined in (3) is continuous.

Proof: Let $f_0 \leq f_1 \leq ...$ be an increasing chain in $R \rightarrow [0,1]^\Omega$. We need to show that

$$\mathcal{R}(\bigsqcup_{n\geq 0} f_n) \;=\; \bigsqcup_{n\geq 0} \mathcal{R}(f_n) \tag{17}$$

For any $r \in R$, we are in one of the following three cases:

1. $r \xrightarrow{\omega}$ for some $\omega \in \Omega$. We have

$$\begin{array}{rcll} \mathcal{R}(\bigsqcup_{n\geq 0} f_n)(r)(\omega) & = & 1 & \text{by (3)} \\ & = & \bigsqcup_{n\geq 0} 1 & \\ & = & \bigsqcup_{n\geq 0} \mathcal{R}(f_n)(r)(\omega) & \\ & = & (\bigsqcup_{n\geq 0} \mathcal{R}(f_n))(r)(\omega) & \end{array}$$

   and

$$\mathcal{R}(\bigsqcup_{n\geq 0} f_n)(r)(\omega') = 0 = (\bigsqcup_{n\geq 0} \mathcal{R}(f_n))(r)(\omega')$$

   for all $\omega' \neq \omega$.

2. $r \not\longrightarrow$. Similar to last case. We have

$$\mathcal{R}(\bigsqcup_{n\geq 0} f_n)(r)(\omega) = 0 = (\bigsqcup_{n\geq 0} \mathcal{R}(f_n))(r)(\omega)$$

   for all $\omega \in \Omega$.

3. Otherwise, $r \xrightarrow{\tau} \Delta$ for some $\Delta \in \mathcal{D}_1(R)$. Then we infer that, for any $\omega \in \Omega$,

$$\begin{array}{rcll} \mathcal{R}(\bigsqcup_{n\geq 0} f_n)(r)(\omega) & = & (\bigsqcup_{n\geq 0} f_n)(\Delta)(\omega) & \text{by (3)} \\ & = & \sum_{r\in\lceil\Delta\rceil} \Delta(r)\cdot(\bigsqcup_{n\geq 0} f_n)(r)(\omega) & \\ & = & \sum_{r\in\lceil\Delta\rceil} \Delta(r)\cdot(\bigsqcup_{n\geq 0} f_n(r))(\omega) & \\ & = & \sum_{r\in\lceil\Delta\rceil} \bigsqcup_{n\geq 0} \Delta(r)\cdot f_n(r)(\omega) & \\ & = & \sum_{r\in\lceil\Delta\rceil} \lim_{n\to\infty} \Delta(r)\cdot f_n(r)(\omega) & \\ & = & \lim_{n\to\infty} \sum_{r\in\lceil\Delta\rceil} \Delta(r)\cdot f_n(r)(\omega) & \text{by Proposition 5.4} \\ & = & \bigsqcup_{n\geq 0} \sum_{r\in\lceil\Delta\rceil} \Delta(r)\cdot f_n(r)(\omega) & \\ & = & \bigsqcup_{n\geq 0} f_n(\Delta)(\omega) & \\ & = & \bigsqcup_{n\geq 0} \mathcal{R}(f_n)(r)(\omega) & \\ & = & (\bigsqcup_{n\geq 0} \mathcal{R}(f_n))(r)(\omega) & \end{array}$$

In the above reasoning, Proposition 5.4 can be applied because we can define the function $f : R \times \mathbb{N} \to \mathbb{R}_{\geq 0}$ by letting $f(r,n) = \Delta(r) \cdot f_n(r)(\omega)$ and check that f satisfies the three conditions in Proposition 5.4. If R is finite, we can extend it to a countable set $R' \supseteq R$ and require $f(r',n) = 0$ for all $r' \in R' \backslash R$ and $n \in \mathbb{N}$.

(a) f satisfies condition **C1**. For any $r \in R$ and $n_1, n_2 \in \mathbb{N}$, if $n_1 \leq n_2$ then $f_{n_1} \leq f_{n_2}$. It follows that

$$f(r,n_1) \;=\; \Delta(r) \cdot f_{n_1}(r)(\omega) \;\leq\; \Delta(r) \cdot f_{n_2}(r)(\omega) \;=\; f(r,n_2).$$

(b) f satisfies condition **C2**. For any $r \in R$, the sequence $\{\Delta(r) \cdot f_n(r)(\omega)\}_{n=0}^{\infty}$ is nondecreasing and bounded by $\Delta(r)$. It follows that the limit $\lim_{n\to\infty} f(r,n)$ exists.

(c) f satisfies condition **C3**. For any $R'' \subseteq R$, the partial sum $\sum_{r\in R''} \lim_{n\to\infty} f(r,n)$ is bounded because

$$\sum_{r\in R''} \lim_{n\to\infty} f(r,n) \;=\; \sum_{r\in R''} \lim_{n\to\infty} \Delta(r) \cdot f_n(r) \;\leq\; \sum_{r\in R''} \Delta(r) \;\leq\; \sum_{r\in R} \Delta(r) \;=\; 1.$$

□

Consequently, the least fixed point of $\mathcal{R}$, i.e. $\mathbb{V}$, is also continuous. Moreover, the function $\mathbb{V}$ can be captured by a chain of approximants. The functions $\mathbb{V}^n,\ n \geq 0$ are defined by induction on n:

$$\mathbb{V}^0(r)(\omega) = 0 \qquad \text{for all } \omega \in \Omega$$
$$\mathbb{V}^{n+1} = \mathcal{R}(\,\mathbb{V}^n\,)$$

Then $\mathbb{V} = \bigsqcup_{n\geq 0}\ \mathbb{V}^n$. This is used in the following result.

Lemma 5.6 Let Δ be a subdistribution in an ω-respecting deterministic pLTS. If $\Delta \Longrightarrow\!\!\!\succ \Delta'$ then $\mathbb{V}(\Delta) = \mathbb{V}(\Delta')$.

Proof: Since the pLTS is ω-respecting we know that $s \xrightarrow{\tau} \Delta$ implies $s \not\xrightarrow{\omega}$ for any ω. Therefore from the definition of the functional $\mathcal{R}$ we have that $s \xrightarrow{\tau} \Delta$ implies $\mathbb{V}^{n+1}(s) = \mathbb{V}^n(\Delta)$, whence by lifting and linearity we get

$$\text{If } \Delta \xrightarrow{\tau} \Delta' \text{ then } \mathbb{V}^{n+1}(\Delta) = \mathbb{V}^n(\Delta') \text{ for all } n \geq 0. \tag{18}$$

Now suppose $\Delta \Longrightarrow\!\!\!\succ \Delta'$. Referring to Definition 4.4 and carrying out a straightforward induction based on (18), we have

$$\mathbb{V}^{n+1}(\Delta) \quad = \quad \mathbb{V}^0(\Delta_{n+1}) + \sum_{k=0}^{n} \mathbb{V}^{n-k+1}(\Delta_k^{\times}) \tag{19}$$

for all $n \geq 0$. This can be simplified further by noting

(i) $\mathbb{V}^0(\Delta)(\omega) = 0$ for every Δ

(ii) $\mathbb{V}^{m+1}(\Delta) = \mathbb{V}(\Delta)$ for every $m \geq 0$, provided Δ is stable.

Applying these remarks to (19) above, since all $\Delta_k^{\times}$ are stable, we obtain

$$\mathbb{V}^{n+1}(\Delta) \quad = \quad \sum_{k=0}^{n} \mathbb{V}(\Delta_k^{\times}) \tag{20}$$

We conclude by reasoning

$$
\begin{aligned}
\mathbb{V}(\Delta) &= \bigsqcup_{n\geq 0} \mathbb{V}^{n+1}(\Delta) \\
&= \bigsqcup_{n\geq 0} \sum_{k=0}^{n} \mathbb{V}(\Delta_k^\times) && \text{from (20) above} \\
&= \bigsqcup_{n\geq 0} \mathbb{V}(\sum_{k=0}^{n} \Delta_k^\times) && \text{by linearity of } \mathbb{V} \\
&= \mathbb{V}(\bigsqcup_{n\geq 0} \sum_{k=0}^{n} \Delta_k^\times) && \text{by continuity of } \mathbb{V} \\
&= \mathbb{V}(\sum_{k=0}^{\infty} \Delta_k^\times) \\
&= \mathbb{V}(\Delta')
\end{aligned}
$$

□

We are now ready to compare the two methods for calculating the set of outcomes associated with a subdistribution:

- using resolutions and the evaluation function $\mathbb{V}$ from page 6.
- using extreme derivatives and the reward function $\$$ from Definition 4.7

Corollary 5.7 In an ω-respecting pLTS $\langle S, \Omega_\tau, \rightarrow\rangle$, the following statements hold.

(i) If $\Delta \Longrightarrow\!\!\succ \Delta'$ then there is a resolution $\langle R, \Theta, \rightarrow_R\rangle$ of Δ such that $\mathbb{V}(\Theta) = \$(\Delta')$.

(ii) For any resolution $\langle R, \Theta, \rightarrow_R\rangle$ of Δ, there exists an extreme derivative Δ' such that $\Delta \Longrightarrow\!\!\succ \Delta'$ and $\mathbb{V}(\Theta) = \$(\Delta')$.

Proof: Suppose $\Delta \Longrightarrow\!\!\succ \Delta'$. By Theorem 5.1, there is a resolution $\langle R, \Theta, \rightarrow_R\rangle$ of Δ with resolving function f and a subdistribution Θ such that $\Theta \Longrightarrow\!\!\succ \Theta'$ and $\Delta' = \text{Img}_f(\Theta')$. By Lemma 5.6, we have $\mathbb{V}(\Theta) = \mathbb{V}(\Theta')$.

Since Θ' and Δ' are extreme derivatives, all the states in their supports are stable. Therefore a simple calculation, using the fact that $\Delta' = \text{Img}_f(\Theta')$, will show that $\mathbb{V}(\Theta') = \$(\Delta')$, from which the required $\mathbb{V}(\Theta) = \$(\Delta')$ follows.

To prove part (ii), suppose that $\langle R, \Theta, \rightarrow_R\rangle$ is a resolution of Δ with resolving function f, so that $\Delta = \text{Img}_f(\Theta)$. We know from Lemma 4.6 that there exists a (unique) subdistribution Θ' such that $\Theta \Longrightarrow\!\!\succ \Theta'$. By Theorem 5.2 we have that $\Delta = \text{Img}_f(\Theta) \Longrightarrow\!\!\succ \text{Img}_f(\Theta')$. The same arguments as in the other direction show that $\mathbb{V}(\Theta) = \$(\text{Img}_f(\Theta'))$. □

6 Agreement of nonnegative and real reward must testing

In this section we prove the agreement of $\sqsubseteq^{\Omega}_{\mathbf{nr}\text{must}}$ with $\sqsubseteq^{\Omega}_{\mathbf{rr}\text{must}}$ for finitary processes, by using failure simulation [2] as a stepping stone. We start with defining the weak action relations $\stackrel{\alpha}{\Longrightarrow}$ for $\alpha \in \mathsf{Act}_\tau$ and the refusal relations $\stackrel{A}{\not\longrightarrow}$ for $A \subseteq \mathsf{Act}$ that are the key ingredients in the definition of the failure simulation preorder.

Definition 6.1 Let Δ and its variants be subdistributions in a pLTS $\langle S, \mathsf{Act}_\tau, \rightarrow\rangle$.

- For $a \in \mathsf{Act}$ write $\Delta \stackrel{a}{\Longrightarrow} \Delta'$ whenever $\Delta \Longrightarrow \Delta^{\text{pre}} \stackrel{a}{\longrightarrow} \Delta^{\text{post}} \Longrightarrow \Delta'$. Extend this to Act_τ by allowing as a special case that $\stackrel{\tau}{\Longrightarrow}$ is simply $\Longrightarrow$, i.e. including identity (rather than requiring at least one $\stackrel{\tau}{\longrightarrow}$).
- For $A \subseteq \mathsf{Act}$ and $s \in S$ write $s \stackrel{A}{\not\longrightarrow}$ if $s \stackrel{\alpha}{\not\longrightarrow}$ for every $\alpha \in A \cup \{\tau\}$; write $\Delta \stackrel{A}{\not\longrightarrow}$ if $s \stackrel{A}{\not\longrightarrow}$ for every $s \in \lceil\Delta\rceil$.
- More generally write $\Delta \Longrightarrow\stackrel{A}{\not\longrightarrow}$ if $\Delta \Longrightarrow \Delta^{\text{pre}}$ for some Δ^{pre} such that $\Delta^{\text{pre}} \stackrel{A}{\not\longrightarrow}$.

Definition 6.2 [Failure simulation preorder] Define $\lhd_{FS}$ to be the largest relation in $S \times \mathcal{D}(S)$ such that if $s \lhd_{FS} \Theta$ then

(i) whenever $\overline{s} \overset{\alpha}{\Longrightarrow} \Delta'$, for $\alpha \in \mathsf{Act}_\tau$, then there is a $\Theta' \in \mathcal{D}(S)$ with $\Theta \overset{\alpha}{\Longrightarrow} \Theta'$ and $\Delta' \;\overline{\lhd_{FS}}\; \Theta'$, and

(ii) whenever $\overline{s} \Longrightarrow\overset{A}{\not\to}$ then $\Theta \Longrightarrow\overset{A}{\not\to}$.

Any relation $\mathcal{R} \subseteq S \times \mathcal{D}(S)$ that satisfies the two clauses above is called a *failure simulation*. Failure simulation preorder $\sqsubseteq_{FS} \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$ is defined by letting $\Theta \sqsubseteq_{FS} \Delta$ whenever there is a Θ^{match} with $\Theta \Longrightarrow \Theta^{\text{match}}$ and $\Delta \;\overline{\lhd_{FS}}\; \Theta^{\text{match}}$.

When only finitary processes are considered, failure simulation preorder is a precongruence relation for pCSP operators, and it is sound and complete for must testing preorder.

Theorem 6.3 [2] For finitary processes P and Q,

(i) If $P \sqsubseteq_{FS} Q$ then for any Ω-test T it holds that $[P \mid_{\mathsf{Act}} T] \sqsubseteq_{FS} [Q \mid_{\mathsf{Act}} T]$.

(ii) $P \sqsubseteq_{FS} Q$ if and only if $P \sqsubseteq^{\Omega}_{\text{pmust}} Q$. □

Because we prune our pLTSs before extracting values from them, we will be concerned mainly with ω-respecting structures. Moreover, we require the pLTSs to be *convergent* in the sense that there is no wholly divergent state s, i.e. $s \Longrightarrow \varepsilon$. A process P is *convergent* if $[\![P]\!]$ generates a convergent pLTS.

Definition 6.4 Let Δ be a distribution in a pLTS $\langle S, \{\omega, \tau\}, \to\rangle$. We write $\mathcal{V}(\Delta)$ for the set of testing outcomes $\{\$\Delta' \mid \Delta \Longrightarrow\!\!\succ \Delta'\}$.

Lemma 6.5 Let Δ and Θ be distributions in an ω-respecting convergent pLTS $\langle S, \Omega_\tau, \to\rangle$. If distribution Δ is stable and $\Delta \;\overline{\lhd_{FS}}\; \Theta$, then $\$\Delta \in \mathcal{V}(\Theta)$.

Proof: We first show that if s is stable and $s \lhd_{FS} \Theta$ then $\$\overline{s} \in \mathcal{V}(\Theta)$. Since s is stable, we have only two cases:

(i) $s \not\to$ Here $\$\overline{s}{=}\vec{0}$, where $\vec{0}(\omega) = 0$ for all $\omega \in \Omega$. Since $s \lhd_{FS} \Theta$ we have $\Theta \Longrightarrow \Theta'$ with $\Theta' \not\to$, whence in fact $\Theta \Longrightarrow\!\!\succ \Theta'$ and $\$\Theta' = \vec{0}$. Thus $\$\overline{s} = \vec{0} \in \mathcal{V}(\Theta)$.

(ii) $s \overset{\omega}{\longrightarrow} \Delta'$ for some Δ' Here $\$\overline{s}{=}\vec{\omega}$, and since $s \lhd_{FS} \Theta$ we have $\Theta \Longrightarrow \Theta' \overset{\omega}{\longrightarrow}$. As the pLTS we are considering is convergent and Θ is a distribution, we know that Θ' is also a distribution. Hence,we have $\$\Theta'{=}\vec{\omega}$. Because the pLTS is ω-respecting, in fact $\Theta \Longrightarrow\!\!\succ \Theta'$ and so again $\$\overline{s} = \vec{\omega} \in \mathcal{V}(\Theta)$.

Now for the general case we suppose $\Delta \;\overline{\lhd_{FS}}\; \Theta$. It is not hard to show that we can decompose Θ into $\sum_{s \in \lceil\Delta\rceil} \Delta(s) \cdot \Theta_s$ such that $s \lhd_{FS} \Theta_s$ for each $s \in \lceil\Delta\rceil$, and recall each such state s is stable. From above we have that $\$\overline{s} \in \mathcal{V}(\Theta_s)$ for those s, and so $\$\Delta = \sum_{\in\lceil\Delta\rceil} \Delta(s) \cdot \$\overline{s} \in \sum_{s \in \lceil\Delta\rceil} \Delta(s) \cdot \mathcal{V}(\Theta_s) = \mathcal{V}(\Theta)$. □

Lemma 6.6 Let Δ and Θ be distributions in an ω-respecting convergent pLTS $\langle S, \Omega_\tau, \to\rangle$. If $\Theta \sqsubseteq_{FS} \Delta$, then it holds that $\mathcal{V}(\Theta) \supseteq \mathcal{V}(\Delta)$.

Proof: Let Δ and Θ be distributions in an ω-respecting convergent pLTS $\langle S, \Omega_\tau, \to\rangle$. We first claim that

(i) If $\Delta \Longrightarrow \Delta'$ then $\mathcal{V}(\Delta') \subseteq \mathcal{V}(\Delta)$.

(ii) If $\Delta \;\overline{\lhd_{FS}}\; \Theta$, then we have $\mathcal{V}(\Delta) \subseteq \mathcal{V}(\Theta)$.

The first claim holds because if $\Delta' \Longrightarrow\!\!\succ \Delta''$ then $\Delta \Longrightarrow \Delta' \Longrightarrow\!\!\succ \Delta''$, i.e. every extreme derivative of Δ' is also an extreme derivative of Δ. For the second claim, we assume $\Delta \;\overline{\lhd_{FS}}\; \Theta$. For any $\Delta \Longrightarrow\!\!\succ \Delta'$ we have the matching transition $\Theta \Longrightarrow \Theta'$ such that $\Delta' \;\overline{\lhd_{FS}}\; \Theta'$. It follows from Lemmas 6.5 and Claim (i) that $\$\Delta' \in \mathcal{V}(\Theta') \subseteq \mathcal{V}(\Theta)$. Consequently, we obtain $\mathcal{V}(\Delta) \subseteq \mathcal{V}(\Theta)$.

Now suppose $\Theta \sqsubseteq_{FS} \Delta$. By definition there exists some Θ' such that $\Theta \Longrightarrow \Theta'$ and $\Delta \;\overline{\lhd_{FS}}\; \Theta'$. By the above two claims we obtain $\mathcal{V}(\Delta) \subseteq \mathcal{V}(\Theta') \subseteq \mathcal{V}(\Theta)$. □

Theorem 6.7 For any finitary convergent processes P and Q, if $P \sqsubseteq_{FS} Q$ then $P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q$.

Proof: We reason as follows.

	$P \sqsubseteq_{FS} Q$	
implies	$[P \mid_{\mathsf{Act}} T] \sqsubseteq_{FS} [Q \mid_{\mathsf{Act}} T]$	Theorem 6.3(i), for any Ω-test T
implies	$\mathcal{V}([P \mid_{\mathsf{Act}} T]) \supseteq \mathcal{V}([Q \mid_{\mathsf{Act}} T])$	$[\cdot]$ is ω-respecting; Lemma 6.6
iff	$\mathcal{A}^{\mathrm{d}}(T,P) \supseteq \mathcal{A}^{\mathrm{d}}(T,Q)$	Definitions 6.4 and 4.8
iff	$\mathcal{A}(T,P) \supseteq \mathcal{A}(T,Q)$	Corollary 5.7 and (4)
implies	$h \cdot \mathcal{A}(T,P) \supseteq h \cdot \mathcal{A}(T,Q)$ for any $h \in [-1,1]^{\Omega}$	
implies	$\prod h \cdot \mathcal{A}(T,P) \leq \prod h \cdot \mathcal{A}(T,Q)$ for any $h \in [-1,1]^{\Omega}$	
iff	$P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q$.	

Note that in the second line above, both $[P \mid_{\mathsf{Act}} T]$ and $[Q \mid_{\mathsf{Act}} T]$ are convergent, since for any convergent process R and very finite process T, by induction on the structure of T, it can be shown that the composition $R \mid_{\mathsf{Act}} T$ is also convergent. □

We are now ready to prove the main result of the paper which states that nonnegative reward must testing is as discriminating as real reward must testing.

Theorem 6.8 For any finitary convergent processes P and Q, it holds that $P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q$ if and only if $P \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}} Q$. □

Proof: The "only if" direction is obvious. Let us consider the "if" direction. Suppose P and Q are finitary processes. We reason as follows.

	$P \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}} Q$	
iff	$P \sqsubseteq^{\Omega}_{\mathrm{pmust}} Q$	Theorem 3.7(ii)
iff	$P \sqsubseteq_{FS} Q$	Theorem 6.3(ii)
implies	$P \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q$.	Theorem 6.7

□

In the presence of divergence, $\sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}}$ is strictly included in $\sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}}$. For example, let P and Q be the processes $\mathsf{rec}\, x.\, x$ and $a.\,\mathbf{0}$, respectively. It holds that $P \sqsubseteq_{FS} Q$ because $P \stackrel{\varepsilon}{\Longrightarrow}$ and the empty subdistribution can failure simulate any processes. It follows from Theorems 6.3(ii) and 3.7(ii) that $P \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{must}} Q$. However, if we apply the test $T = a.\omega$ and reward tuple h with $h(\omega) = -1$, then

$$\begin{array}{lllllll} \prod h \cdot \mathcal{A}^{\mathrm{d}}(T,P) & = & \prod h \cdot \{\varepsilon\} & = & \prod\{0\} & = & 0 \\ \prod h \cdot \mathcal{A}^{\mathrm{d}}(T,Q) & = & \prod h \cdot \{\vec{\omega}\} & = & \prod\{-1\} & = & -1 \end{array}$$

As $\prod h \cdot \mathcal{A}^{\mathrm{d}}(T,P) \not\leq \prod h \cdot \mathcal{A}^{\mathrm{d}}(T,Q)$, we see that $P \not\sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{must}} Q$.

7 Conclusion

We have introduced a notion of real reward testing which extends the traditional nonnegative reward testing with negative rewards. It turned out that real reward may preorder is the inverse of real reward must preorder, and vice versa. More interestly, for finitary convergent processes, real reward must testing preorder coincides with nonnegative reward testing preorder. In order to prove this result, we have presented two testing approaches and shown their coincidence, which involved proving some analytic properties such as the continuity of a function for calculating testing outcomes, and bounded continuity of a class of binary functions.

Below we comment on nonnegative reward may testing. Let P and Q be two finitary convergent processes. Using results from [2] we know that $P \sqsubseteq_{FS} Q$ implies $Q \sqsubseteq_{S} P$. In view of Figure 1 we obtain that $Q \sqsubseteq^{\Omega}_{\mathbf{rr}\mathrm{may}} P$ implies $Q \sqsubseteq^{\Omega}_{\mathbf{nr}\mathrm{may}} P$. In other words, real reward may testing is more discriminating than nonnegative reward may testing without the presence of divergence.

References

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

[2] Y. Deng, R.J. van Glabbeek, M. Hennessy & C.C. Morgan (2009): *Testing finitary probabilistic processes*. In Proc. *CONCUR'09*, LNCS 5710, Springer, pp. 274–288.

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

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

[5] Y. Deng, R.J. van Glabbeek, C.C. Morgan & C. Zhang (2007): *Scalar outcomes suffice for finitary probabilistic testing*. In Proc. *ESOP'07*, LNCS 4421, Springer, pp. 363–378.

[6] R. De Nicola & M. Hennessy (1984): *Testing equivalences for processes*. *Theoretical Computer Science* 34, pp. 83–133.

[7] C.A.R. Hoare (1985): *Communicating Sequential Processes*. Prentice-Hall.

[8] B. Jonsson, C. Ho-Stuart & Wang Yi (1994): *Testing and refinement for nondeterministic and probabilistic processes*. In Proc. *FTRTFT'94*, LNCS 863, Springer, pp. 418–430.

[9] B. Jonsson & Wang Yi (1995): *Compositional testing preorders for probabilistic processes*. In Proc. *LICS'95*, IEEE Computer Society Press, pp. 431–441.

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

[11] E.-R. Olderog & C.A.R. Hoare (1986): *Specification-oriented semantics for communicating processes*. *Acta Informatica* 23, pp. 9–66.

[12] M. Puterman (1994): *Markov Decision Processes*. Wiley.

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

[14] R. Segala (1996): *Testing probabilistic automata*. In Proc. *CONCUR'96*, LNCS 1119, Springer, pp. 299–314.

[15] Wang Yi & K.G. Larsen (1992): *Testing probabilistic and nondeterministic processes*. In Proc. *PSTV'92*, *IFIP Transactions* C-8, North-Holland, pp. 47–61.