Bisimulation Through Probabilistic Testing
(Preliminary Report)

Kim G. Larsen
Department of Mathematics and Computer Science
Aalborg University Center
9000 Aalborg, Denmark.

Arne Skou
The University of Sussex
Computer Science
Brighton BN1 9QH, Sussex, England.

Abstract

We propose a language for testing concurrent processes and examine its strength in terms of the processes that are distinguished by a test. By using probabilistic transition systems as the underlying semantic model, we show how a testing algorithm with a probability arbitrary close to 1 can distinguish processes that are not bisimulation equivalent. We also show a similar result (in a slightly stronger form) for a new process relation called $\frac{2}{3}$-bisimulation — lying strictly between that of simulation and bisimulation. Finally, the ultimately strength of the testing language is shown to identify an even stronger process relation, called probabilistic bisimulation.

1 Motivation

Since the appearance of Milners observational view of process behaviour, [Mil 80] much work has been devoted to the development of theories, that support the specification and verification of parallel systems (in particular, compositional theories have been sought). In spite of the extreme importance of this work, there are several reasons which may justify work on the more practical question of testing an implementation against its specification:

- Verification of large systems is at present too costly and time consuming in most situations.
- Implementations are frequently produced using some ad hoc programming language, which excludes a formal verification.
- Most people do not build their own systems, rather they buy them from some dealer who obviously will make various claims about the abilities of the product. Normally, the dealer will give no information as to how the system is build, and it is therefore plain impossible for the buyer to verify whether or not the claims hold. Instead, the buyer will be given some amount of time for testing the system after which he must decide whether to keep it (i.e. he believes the claims) or return it (he does not believe the claims).
- New insight may be gained by looking at concurrency from an alternative (testing) viewpoint.

We consider a test as the description of an algorithm (guaranteed to terminate within some prescribed amount of time) for how to experiment on a machine (process) equipped with buttons. During the execution of a particular test the experimenter (or observer) tries to press one button at a time as prescribed by the algorithm, and each time he notes whether it goes down or not (success or failure).

This view of a test accords with previous work on the subject, but there are some remarkable differences in the way the process is controlled during test execution. Hennessy and DeNicola [DeNicHen 84] modestly assumes that the test can proceed as long as success is reported, while Phillips [Phil 87] allows it to continue in case of failure. These two approaches are included in the framework of Abramsky [Abr 88], who furthermore requires the ability to take multiple copies of the process at any stage of the test in order to experiment on one copy at a time. As a special option in this feature he also demands that the process can be forced (by the observer) to enumerate all its possible (non-deterministic) transitions under any button, thereby enabling the test to be exhaustive. This very last option Milner calls 'controlling the weather conditions' [Mil 81], and it makes it possible to test if two processes are bisimulation equivalent [Park 80, Mil 83] (or rather, whenever two processes are not bisimulation equivalent, there exists a test distinguishing them).

In our view the ability "to control the weather conditions" in the above sense is in direct conflict with the observational viewpoint of process behaviour: no real system is equipped with such a "weather control" knob. Therefore we consider this feature as an unrealistic one, and as a consequence rule it out. On the other hand we accept the copying feature because it in many situations can be realized by a simple core dump procedure and also because it is an applied procedure

in several kinds of fault tolerant systems.

The 'weather control' (or global testing), and the bisimulation notion has been criticised by Bloom, Estrail and Meyer [BloIstrMey 88] from the point of view that it cannot be captured as a trace congruence of any 'reasonable' process constructions. We agree that the global testing is not a realistic assumption. However, the conclusion of [BloIstrMey 88] on the untraceability of bisimulation we consider too strong. Actually we show that any difference between two non–bisimular processes can be detected (with a high probability) by some test. This result is achieved by applying a finer underlying semantic model for processes, namely that of probabilistic transition systems.

Probabilistic processes or programs have been studied before, but to the best of our knowledge never in the context of testing. Pnueli and Zuck [PnuZuck 86] has studied the formal verification of probabilistic distributed programs based on temporal logic; Kozen has given a semantic interpretation of probabilistic programs using generalizations of Scott Domains [Koz 81], and introducesd a Probabilistic Propositional Dynamic Logic useful for calculating the expected running time of probabilistic programs; Vardi has examined the possibility of automatic verification of probabilistic processes in [Vardi 85].

The present paper is organized as follows. In section 2 the probabilistic model is presented together with a simple test language. This induces another language for writing down the experiences that may be observed during an execution of a test, and as a consequence of applying the probabilistic model, each process defines a probability density function over the observation language for a given test.

The main result of this paper is a systematic framework for testing a process against its specification. We consider a specification as being formed by a number of desired properties of the final implementation. Such properties may be formulated in e.g. modal logic, temporal logic or process algebra. In section 3 we introduce the notion of *testable property* as a property which through testing may correctly be decided to hold of processes with arbitrarily high probability. In the remaining sections the properties of three different modal logics are shown to be testable. One of the logics is the well known Hennessy-Milner logic [HenMil 85] and the other two are the restriction of HML, called Limited Modal logic in [BloIstrMey 88], and a new Probabilistic Modal logic. For each logic we also give an operational definition of the associated process equivalence. One of these is of course the usual bisimulation relation whereas the finest is a new relation called probabilistic bisimulation.

2 Probabilistic Transition Systems. A Simple Test Language

In order to describe a test as an algorithm running on a process, we adopt the well-established notion of labelled transition systems [Plotkin 81] as a tool for defining the operational behaviour of processes. This model has been extensively used over the last few years for describing properties of communicating processes and for defining relations under which such processes are to be considered indistinguishable [Pnue 85]. Bisimulation is one such relation preserving deadlock properties, and it has been argued by Milner [Mil 81] that one has to control the non-determinism (i.e. the weather control), of processes, if non-bisimilarity is going to be detectable by a test. As we find the 'weather control' unrealistic, the model must be refined instead, and one immediate refinement is to consider each transition happening with some fixed probability according to the definition below. Thus, even though the observer cannot himself control the weather, he can now — due to the probabilistic nature of transitions — with arbitrarily high degree of confidence, assume that all transitions have been examined, simply by repeating the experiment many times (using of course the copying facility).

Definition 2.1 *A probabilistic transition system is a tuple*

$$\mathcal{P} = (Pr, Act, Can, \mu)$$

where Pr is a set of processes (or states), Act is the set of (observable) actions that processes may perform, Can is an Act-indexed family of sets of processes, with Can_a indicating the set of processes that can perform the action a, μ is a family of probability distributions, $\mu_{p,a} : Pr \to [0,1]$, for any $a \in Act$ and $p \in Can_a$, indicating the possible next states (and their probabilities) after p has performed a.

Note that whenever $p \in Can_a$ we have $\sum_{p'} \mu_{p,a}(p') = 1$ since $\mu_{p,a}$ is a probability distribution. Informally, $\mu_{p,a}(p') = \mu$ may be read as "p can perform the action a and with probability μ become the process p' afterwards". Thus p' is a possible next state after having performed a on p just in case $\mu_{p,a}(p') > 0$. We shall in the remainder of this paper use the following notations:

$p \xrightarrow{a}_{\mu} p'$	whenever $p \in Can_a$ and $\mu_{p,a}(p') = \mu$
$p \xrightarrow{a} p'$	whenever $p \xrightarrow{a}_{\mu} p'$ for some $\mu > 0$
$p \xrightarrow{a}$	whenever $p \in Can_a$
$p \not\xrightarrow{a}$	whenever $p \notin Can_a$

Also, we shall assume that there is a lower limit as to the probability of transitions (referred to in the following as the *minimal probability assumption*): i.e. we assume the existence of some $\varepsilon > 0$ such that whenever $p \xrightarrow{a}_{\mu} p'$, then either $\mu = 0$ or $\mu \geq \varepsilon$. Clearly this implies that all process are finitely branching under $\xrightarrow{a}$ for any action a (in fact $\lceil \frac{1}{\varepsilon} \rceil$ is a universal upper limit on the branching), a condition also known as *image-finiteness* [HenMil 85]. Figure 1 gives examples of probabilistic transition systems.

The execution of a test algorithm on a process basically consists of a series of button pressures (i.e. attempts to observe an action) and as argued in the introduction such an attempt may either be issued on the current process state or on a 'fresh' process copy obtained earlier. These basic testing capabilities are reflected in the following simple test language:

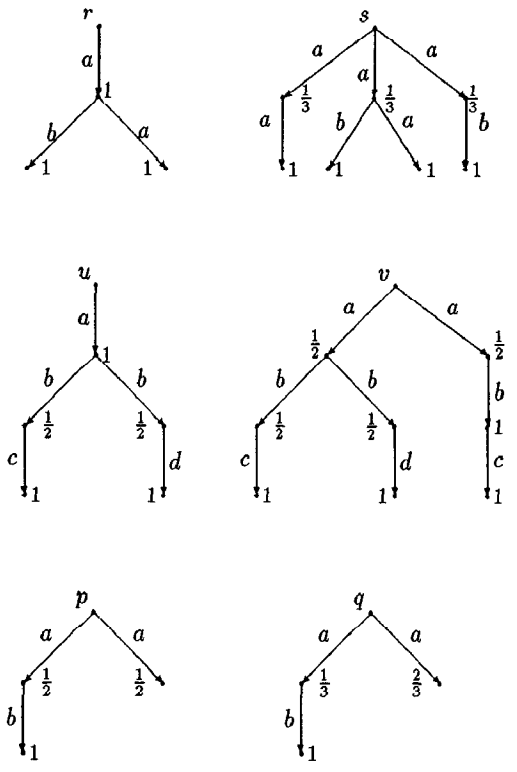

Figure 1: Probabilistic processes

Definition 2.2 *The testing language T has the following syntax*

$$t ::= \omega \mid a.t \mid (t_1, \ldots, t_n),$$

where ω is a special symbol for termination and $a \in Act$. If all elements in a tuple test are identical we use the shorthand $(t)^n$.

A test specifies an algorithm for how an observer shall experiment on a process (i.e. which buttons to press when). Informally ω is the test which requires no experiment at all (and therefore will yield no information); $a.t$ describes a test consisting in first applying pressure on the a-button and in case of success proceeding with t. $(t_1, \ldots, t_n)$ requires that n copies of the current state is taken allowing all the tests $t_1, \ldots, t_n$ to be performed independently on the same state. We shall not here give any formal operational semantics of tests, but rather focus on the concept of a *test observation*. During the execution of a particular test on a process, the observer is subject to a series of experiences, each consisting of success or failure of pressing a button. These experiences are written down by the observer according to the (syntactic) structure of the test, and at the end of the execution a full description of all the experiences is available in the form of an *observation*. Obviously — because of the inherent non-determinism of the processes — there may in general be many possible resulting observations when executing a test on some process. We define the set of observations that may possibly be obtained from a particular test (on any process) structurally on T as follows:

test	*observation set*
$a.b.\omega$	$\{0_a, 1_a : E(b)\}$
$a.(a.\omega, b.\omega)$	$\{0_a, 1_a : E(a) \times E(b)\}$
$a.b.(c.\omega, d.\omega)$	$\{0_a, 1_a : \{0_b, 1_b : E(c) \times E(d)\}\}$

Figure 2. Tests and associated observation set. $(E(x) = \{0_x, 1_x : \{1_\omega\}\})$

Definition 2.3 *A test t induces the following observation set O_t*

$$\begin{aligned} O_\omega &= \{1_\omega\} \\ O_{a.t} &= \{0_a\} \cup \{1_a : e \mid e \in O_t\} \\ O_{t_1,\ldots,t_n} &= O_{t_1} \times \cdots \times O_{t_n} \end{aligned}$$

The observation set for the terminating test ω is just a singleton set. Thus ω provides no basis for distinguishing processes. An observation of a test $a.t$ is either 0_a, indicating that the process did not respond to a (thus the test terminates in this case), or of the form $1_a : e$, where 1_a indicates positive response on a and e is an observation of the following test t. An observation of a tuple test $(t_1, \ldots, t_n)$ consists of tuples of observation $(e_1, \ldots, e_n)$ with $e_i \in O_{t_i}$.

Figure 2 contains three tests and their associated observation sets (using a slightly liberal notation for sets, with $1_a : U$ respectively $\{0_a, U\}$ abbreviating $\{1_a : e \mid e \in U\}$ respectively $\{0_a\} \cup U$).

The execution of a given test t on a particular process p results in some observation within O_t. However, when p is non-deterministic, there may be many possible resulting observations. In our model the non-determinism within p is modelled probabilistic. Thus the possible resulting observations will occur with different probabilities. In fact any process p defines a probability distribution $P_{t,p}$ on O_t as follows:

Definition 2.4 *Let p be a process and t a test. Then $P_{t,p} : O_t \to [0,1]$ is the probability distribution defined structurally on t as follows:*

1. $P_{\omega,p}(1_\omega) = 1$
2. $P_{a.t,p}(0_a) = \begin{cases} 1 & \text{if } p \not\xrightarrow{a} \\ 0 & \text{otherwise} \end{cases}$

   $P_{a.t,p}(1_a : e) = \begin{cases} 0 & \text{if } p \xrightarrow{a} \\ \sum_{p'} \mu_{p,a}(p') \cdot P_{t,p'}(e) & \text{otherwise} \end{cases}$

   where $e \in O_t$
3. $P_{(t_1,\ldots,t_n),p}((e_1, \ldots, e_n)) = \prod_i P_{t_i,p}(e_i)$ *where* $\forall i. e_i \in O_{t_i}$

It is easily proved that for any p and t, $\sum_{e \in O_t} P_{t,p}(e) = 1$ ensuring that the definition above indeed is that of a probability distribution.

In 2, the probability of observing 0_a clearly depends on whether p can perform a or not. If p can perform a, this observation is impossible, and otherwise it is the only one possible. When p cannot perform a, clearly no observation of the form $1_a : e$ is possible. Otherwise the probability is

$t = a.b.\omega$		
observation e	$P_{t,p}(e)$	$P_{t,q}(e)$
0_a	0	0
$1_a : 0_b$	$\frac{1}{2}$	$\frac{2}{3}$
$1_a : 1_b : 1_\omega$	$\frac{1}{2}$	$\frac{1}{3}$

$t = a.(a.\omega, b.\omega)$		
observation e	$P_{t,r}(e)$	$P_{t,s}(e)$
0_a	0	0
$1_a : (0_a, 0_b)$	0	0
$1_a : (0_a, 1_b)$	0	$\frac{1}{3}$
$1_a : (1_a, 0_b)$	0	$\frac{1}{3}$
$1_a : (1_a, 1_b)$	1	$\frac{1}{3}$

$t = a.b.(c.\omega, d.\omega)$		
observation e	$P_{t,u}(e)$	$P_{t,v}(e)$
0_a	0	0
$1_a : 0_b$	0	0
$1_a : 1_b : (0_c, 0_d)$	0	0
$1_a : 1_b : (0_c, 1_d)$	$\frac{1}{2}$	$\frac{1}{4}$
$1_a : 1_b : (1_c, 0_d)$	$\frac{1}{2}$	$\frac{1}{4}$
$1_a : 1_b : (1_c, 1_d)$	0	0

Figure 3. Tables showing $P_{t,p}(e)$ for various p, t and $e \in O_t$.

the sum over all processes, where each process contributes with the probability of observing e, but weighted with the probability of the processes being the next state after having performed a.

In 3, assuming independance of the testing on the n copies of p, the probability of a tuple observation is simply the product of the probabilities of the component observations.

Figure 3 shows for three tests (figure 2) the probability of the observations with respect to various processes (figure 1).

3 Testing properties of processes

We want to use tests for deciding whether a particular implementation is correct wrt. a given specification. This question is of a very practical nature and traditionally there has been a strong distinction between a test and a proof in the sense that it is commonly said that:

> *Correctness cannot be established through testing. Testing can only detect errors, but never exclude errors.*

Although this statement is clearly true, it is also generally believed that the more tests a system passes, the more confidence we may have in the the correctness of the system. In fact we may hope that extensive testing can confirm the correctness of a system with arbitrary confidence in the following way:

We view a specification as a number of desired properties of the final implementation. Such properties may be described in a number of ways (e.g. modal logic, temporal logic, process algebra) but the important thing is that a property divides the set of possible implementations in two: those enjoying (satisfying) the property and those that don't.

Thus, in the following a property Φ will simply be a set of processes, and we consider an implementation p as being correct with respect to Φ just in case $p \in \Phi$. We want to settle this question of correctness by executing a test on p, and there *may* be situations where the observation e resulting from such a test decides the question completely, but it is more likely that the observation leaves the question open. This situation is of course not satisfactory.

However, it will be an improvement if we can find a test t_Φ especially made for Φ, such that certain observations E_Φ (called the set of evidence for Φ) occur with very high probability in case $p \in \Phi$, but with very low probability if $p \notin \Phi$. If such a test exists and the observation e resulting from running t_Φ on p is in E_Φ, then we feel fairly confident that $p \in \Phi$ and similarly that $p \notin \Phi$ if e is not in E_Φ. We consider a property as being *testable* if a test along these lines exists for any desired degree of confidence, and the following definition makes this precise:

Definition 3.1 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system. Then a property $\Phi \subseteq Pr$ is* testable *iff for any $\delta > 0$ there exists a test t_Φ and an observation set $E_\Phi \subseteq O_{t_\Phi}$ such that the following holds for any process $p \in Pr$:*

1. *Whenever $p \in \Phi$, then $P_{t_\Phi,p}(E_\Phi) \geq 1 - \delta$*
2. *Whenever $p \notin \Phi$, then $P_{t_\Phi,p}(E_\Phi) \leq \delta$*

where $P_{t_\Phi,p}(E_\Phi) = \sum_{e \in E_\Phi} P_{t_\Phi,p}(e)$.

We shall refer to δ as the *level of significance*, and intuitively it gives an upper bound of making a wrong decision, i.e. concluding that p enjoys Φ, when it does not, or dually concluding that p does not satisfy Φ when in fact it does.

To relate our terminology with the one normally used within Hypothesis Testing in statistics [CoxHin 74], we may consider the assumption "p enjoys Φ" as a *null hypothesis* H_0, and the complement of E_Φ as the *critical region* for the test t_Φ. We shall not continue the analogy any further, but just mention the strong resemblance between testability of Φ and what might be called *consistency* of a test within statistics.

In the same way we consider a class (collection) of properties as being testable if all its members are testable:

Definition 3.2 *A class C of properties is testable iff for all $\Phi \in C$, Φ is testable*

The crucial question is then what classes are testable in the above sense and also given a testable class, which processes are equivalent in the sense that no property in the class separates them.

In the remainder of this paper we consider three property classes defined in terms of various modal logics. All three

classes are shown to be testable in the above sense, and the associated equivalences on processes are given operational chartacterizations.

The first and weakest property is given by the restriction of Hennessy-Milner Logic (HML) called *Limited Modal Logic* in [BloIstrMey 88]. It is shown that two processes are indistinguishable with respect to properties in this class, just in case they define the same sets of possible observations for all tests. Operationally, the equivalence is characterized by a new notion of $\frac{2}{3}$-*bisimulation*, a relation between processes lying strictly between that of simulation [Lar 86] and bisimulation [Park 80, Mil 83].

The second class of properties, which we show to be testable, is that given by Hennessy-Milner Logic [HenMil 85]. This shows — in contrast to the conclusion reached by [BloIstrMey 88] — that non-bisimular processes can indeed be distinguished "at the terminal", at least when our finer probabilistic model is adopted. The operational characterization of HML is of course obtained through *bisimulation* as shown in [HenMil 85].

The last and strongest class of properties is given by a *Probabilistic Modal Logic* (PML) with the two (next-state) modalities $\langle a\rangle$ and $[a]$ of HML having been replaced by a continuum of (next-state) modalities $\langle a\rangle_\mu$, where $0 \leq \mu \leq 1$. It is shown that two processes are indistinguishable under PML, just in case they give the exact same probability distribution on the observation set of *any* test. Thus, if two processes are indistinguishable with respect to PML, they are in fact indistinguishable with respect to *any* testable property, and could be called *test equivalent*. Operationally, the equivalence is characterized by a notion of *probabilistic bisimulation*.

4 Limited Modal Logic and 2/3–Bisimulation

First let us recall the well-known Hennessy-Milner Logic introduced in [HenMil 85].

Definition 4.1 *The formulas of HML are given by the following syntax:*

$$F ::= \mathrm{tt} \mid \mathrm{ff} \mid [a]F \mid \langle a\rangle F \mid F_1 \wedge F_2 \mid F_1 \vee F_2$$

The satisfaction relation, $p \models F$, between processes and formulas is defined as usual for modal logics and Kripke models (see [HughCres 68]). Thus, $p \models \langle a\rangle F$, whenever $p' \models F$ for some p' with $p \xrightarrow{a} p'$, and dually, $p \models [a]F$, whenever $p \xrightarrow{a} p'$ implies $p' \models F$.

We shall view any HML formula F as the property (i.e. a set of processes), consisting of all the processes satisfying it.

The fundamental result of [HenMil 85] is that two processes will satisfy exactly the same HML formulas just in case they are bisimular (provided $\longrightarrow$ is image-finite).

Definition 4.2 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system. Then a* bisimulation *$\mathcal{R}$ is a binary relation on Pr such that whenever $p\mathcal{R}q$ and $a \in Act$ then the following holds:*

1. *Whenever $p \xrightarrow{a} p'$, then $q \xrightarrow{a} q'$ such that $p'\mathcal{R}q'$*
2. *Whenever $q \xrightarrow{a} q'$, then $p \xrightarrow{a} p'$ such that $p'\mathcal{R}q'$*

Two processes p and q are said to be bisimular *in case (p, q) is contained in some bisimulation R. We write $p \sim q$ in this case.*

Before proving the testability of HML itself, let us consider the restriction, Limited Modal Logic (LML), introduced in [BloIstrMey 88].

Definition 4.3 *The formulas of LML are given by the following syntax:*

$$F ::= \mathrm{tt} \mid [a]\mathrm{ff} \mid \langle a\rangle F \mid F_1 \wedge F_2$$

The main limitation of LML is that the formulas only allow a very restrictive use of $[a]$, just enabling the logic to express deadlock on actions. Also (in contrast to the version of LML given in [BloIstrMey 88]), formulas of LML cannot contain neither ff or $\vee$. However, it is easily proved that the addition of ff and $\vee$ to (our version of) LML does not increase its distinguishing power.

The operational characterization of LML is obtained by the following notion of $\frac{2}{3}$–bisimulation:

Definition 4.4 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system. Then a $\frac{2}{3}$–bisimulation $\mathcal{R}$ is a binary relation on Pr such that whenever $p\mathcal{R}q$ and $a \in Act$ then the following holds:*

1. *Whenever $p \xrightarrow{a} p'$, then $q \xrightarrow{a} q'$ for some q' with $p'\mathcal{R}q'$*
2. *Whenever $q \xrightarrow{a}$, then $p \xrightarrow{a}$*

Two processes p and q are said to be $\frac{2}{3}$–bisimular just in case (p, q) are contained in some $\frac{2}{3}$–bisimulation and likewise (q, p). We write $p \simeq q$ in this case.

From the above definition it is clear that the notion of bisimulation is strictly stronger than that of $\frac{2}{3}$–bisimulation (in figure 1, $u \simeq v$ but $u \not\sim v$). Also, the notion of $\frac{2}{3}$–bisimulation is strictly stronger than that of simulation (being 'half' of bisimulation, c.f. [Lar 86]) (in figure 1, $r \leq s$ and $s \leq r$ but $s \not\simeq r$).

Now, LML characterizes $\frac{2}{3}$–bisimulation in the following sense

Theorem 4.5 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system satisfying the minimal probability assumption. Then two processes are $\frac{2}{3}$–bisimular just in case they satisfy exactly the same LML-formulas.*

In figure 1, we see that $s \models \langle a\rangle[b]\mathrm{ff}$, whereas $r \not\models \langle a\rangle[b]\mathrm{ff}$. Thus , it follows from the above theorem that $s \not\simeq r$.

In order to establish the testability of LML, we introduce properties of the form $[t, e]$ (t being a test and $e \in O_t$), consisting of the processes p for which $P_{t,p}(e) > 0$. Thus p will satisfy $[t, e]$ iff e is a possible resulting observation when executing on p.

Now, the key fact which will lead to the testability of LML, is that any property described as an LML formula may alternatively be described on the form $[t, e]$, and vice versa.

Lemma 4.6

A *Let F be an LML formula. Then there exists a test τ_F and an observation $e_F \in O_{\tau_F}$ such that $p \models F$ iff $p \in [\tau_F, e_F]$.*

B *Let t be a test and e an observation of t. Then there exists a formula $F_{t,e}$ such that $p \models F_{t,e}$ iff $p \in [t, e]$.*

PROOF: We only give the constructive definitions of $[\tau_F, e_F]$ and $F_{t,e}$.

A $[\tau_F, e_F]$ is defined by structure on F as follows:

$$\begin{aligned}
[\tau_{\mathrm{tt}}, e_{\mathrm{tt}}] &= [\omega, 1_\omega] \\
[\tau_{\langle a\rangle F}, e_{\langle a\rangle F}] &= [a.\tau_F, 1_a : e_F] \\
[\tau_{[a]\mathrm{ff}}, e_{[a]\mathrm{ff}}] &= [a.\omega, 0_a] \\
[\tau_{F_1 \wedge F_2}, e_{F_1 \wedge F_2}] &= [(\tau_{F_1}, \tau_{F_2}), (e_{F_1}, e_{F_2})]
\end{aligned}$$

B $F_{t,e}$ is defined by structure on t as follows:

$$\begin{aligned}
F_{\omega, 1_\omega} &= \mathrm{tt} \\
F_{a.t, 0_a} &= [a]\mathrm{ff} \\
F_{a.t, 1_a : e} &= \langle a\rangle F_{t,e} \\
F_{(t_1, \ldots, t_n)} &= \wedge_i F_{t_i}
\end{aligned}$$

□

From the above lemma and the characterization theorem 4.5, it follows that two (image-finite) processes are $\frac{2}{3}$–bisimular just in case they assign non-zero probabilities to exactly the same observations of any test. Thus, it follows from the tables of figure 3 that $r \not\simeq s$.

Now, let F be a property described in LML and let $[\tau_F, e_F]$ be its alternative description according to lemma 4.6. We may then test the property F in the following way: the test t_F will be of the form $(\tau_F)^N$, (where N is to be determined by the desired level of significance), and the evidence set E_F consists of all observations $(e_1, \ldots, e_N)$ with e_F as a component. By increasing N, clearly the probability of obtaining an observation within E_F — when indeed $p \models F$ — can be made arbitrarily high. On the other hand, the probability of obtaining an observation within e_F will — regardless of N — be 0 when $p \not\models F$.

Under the minimal probability assumption, it is easily verified that in fact $P_{\tau_F, p}(e_F) \geq \varepsilon^{|F|}$ whenever $p \models F$, where $|F|$ is the 'size' of F (defined inductively by $|\mathrm{tt}| = |[a]\mathrm{ff}| = 0, |\langle a\rangle F| = 1 + |F|$ and $|F_1 \wedge F_2| = |F_1| + |F_2|$).

Thus, for a given N, $P_{t_F, p}(E_F) \geq 1 - (1 - \varepsilon^{|F|})^N$. Hence, in order to meet some specified level of significance δ, we may simply choose N so that $(1 - \varepsilon^{|F|})^N \leq \delta$.

We can now assert the testability of LML in a slightly stronger form:

Theorem 4.7 *Let F be an LML formula, and $\delta > 0$ a desired level of significance. Then there exists a test t_F and an evidence set $E_F \subseteq O_{t_F}$ such that*

1. *Whenever $p \models F$, then $P_{t_F, p}(E_F) \geq 1 - \delta$*
2. *Whenever $p \not\models F$, then $P_{t_F, p}(E_F) = 0$*

Note, that when testing for LML properties, we will never conclude that a process satisfies a property when in fact it does not.

Example 4.8 *Let $F = \langle a\rangle[b]\mathrm{ff}$, the level of significance $\delta = 0.1$ and $\varepsilon = \frac{1}{3}$. Then $\tau_F = a.b.\omega$ and $e_F = 1_a : 0_b$. Now, $(1 - \varepsilon^{|F|})^N \leq \delta$ holds for $N \geq 6$. Thus, $t_F = (\tau_F)^6$, and calculations show that $P_{t_F, s}(E_F) \approx 0.92$, whereas $P_{t_F, r}(E_F) = 0$, where s and r are processes from figure 1.*

5 Hennessy–Milner Logic and Bisimulation

Having established the testability of LML, we now face the full class of HML formulas.

The question of testability of HML boils down to showing how to test $[a]F$ formulas in general. Assume (inductively) that t_F is a test for F with evidence set E_F (with respect to some level of significance δ_F), then $a.t_F$ seems like a suitable basis for testing $[a]F$. However, $a.t_F$ itself will only examine a single a–transition of processes, and the chance of erroneously concluding that a processes satisfies $[a]F$, when in fact it does not, may consequently be high. In order to make our conclusion more safe, we must repeat the test $a.t_F$, so that we with high probability can assume that all a–transitions have been examined.

Thus, to test for the property $[a]F$, we propose a test of the form $t_{[a]F} = (a.t_F)^N$, where N is to be determined by the desired level of significance. The set of evidence $E_{[a]F}$ consists of $(0_a, \ldots, 0_a)$ (indicating that the process can not perform a, and therefore satisfies $[a]F$) together with all observations $(1_a : e_1, \ldots, 1_a : e_N)$, where *all* e_i confirms F (i.e. $e_i \in E_F$). With this choice of $t_{[a]F}$ and $E_{[a]F}$, the probability of getting evidence for non–satisfying processes will clearly approach 0 for $N \longrightarrow \infty$. To illustrate this fact, consider the proces p of figure 1, and let $F = [a]\langle b\rangle\mathrm{tt}$, a property clearly *not* satisfied by p. Then for $t_F = (a.b.\omega)^N$, $P_{t_F, p}(E_F) = (\frac{1}{2})^N \longrightarrow 0$ as $N \longrightarrow \infty$.

Now, let us turn to the problem of testing $\langle a\rangle F$ formulas. Assume (inductively) that t_F is a test for F with evidence set E_F (with respect to some level of significance δ_F). Also in this case, $a.t_F$ seems like a suitable test candidate. However, since $a.t_F$ only examines a single a–transition of processes, we may now erroneously conclude that a process does not satisfy $\langle a\rangle F$ when in fact it does, simply because a "wrong" a–transition was examined. Again, repeating $a.t_F$ will resolve the problem.

Thus, to test a property $\langle a\rangle F$, we use a test of the form $t_{\langle a\rangle F} = (a.t_F)^N$, where N has to be determined from the desired level of significance. However, in this case (in contrast to that of $[a]F$) the set of evidence $E_{\langle a\rangle F}$ consists of the observations $(1_a : e_1, \ldots, 1_a : e_N)$ with *at least one* e_i confirming F (i.e. $e_i \in E_F$). It is clear that these choices will make the probability of getting evidence for satisfying processes approach 1 as N increases. *Unfortunately*, so does the probability of getting evidence for *non–satisfying* processes. For p not satisfying $\langle a\rangle F$ it is easily shown that $P_{t_{\langle a\rangle F}, p}(E_{\langle a\rangle F})$ is bounded above by $1 - (1 - \delta_F)^N$. However,

$1-(1-\delta_F)^N \longrightarrow 1$ as $N \longrightarrow \infty$, so this upper bound does not establish the testability of $\langle a\rangle F$. The way we resolve this discrepancy in the proof is to reverse the order in which δ_F and N are determined. Thus, let the desired level of significance δ be given. Then — assuming $\delta_F = \frac{1}{2}$ say — we first choose N so that $P_{t_{\langle a\rangle F},p}(E_{\langle a\rangle F}) \geq 1-\delta$, whenever p satisfies $\langle a\rangle F$. Having determined N, we now fix δ_F so that when p does not satisfy $\langle a\rangle F$, $P_{t_{\langle a\rangle F},p}(E_{\langle a\rangle F}) \leq 1-(1-\delta_F)^N \leq \delta$. Clearly this may be done as $(1-\delta_F)^N \longrightarrow 1$ for $\delta_F \longrightarrow 0$. It is easy to see that this lowering of δ_F does not decrease $P_{t_{\langle a\rangle F},p}(E_{\langle a\rangle F})$ for p satisfying $\langle a\rangle F$. Based on these informal arguments we now state the following main theorem:

Theorem 5.1 *The formulas of Hennessy-Milner Logic are testable.*

Example 5.2 *Consider the processes u and v of figure 1. Clearly $u \not\sim v$, and in fact $F = \langle a\rangle[b](\langle c\rangle \mathrm{tt} \wedge [d]\mathrm{ff})$ is a distinguishing property (satisfied by v but not u). The test for F is of the form $t = (a.(b.(c.\omega, d.\omega))^{N_2})^{N_1}$ with the evidence set E containing tuppels $(e_1, \ldots, e_{N_1})$ with $e_i = 1_a : (1_b : (1_c : 1_\omega, 0_d))^{N_2}$ or $e_i = 1_a : (0_b)^{N_2}$ for some component e_i. Taking $\delta = 0.2$ (a very modest level of significance), and $\varepsilon = \frac{1}{2}$, the constructions in the proof of theorem 5.1 yields $N_1 = 6$ and $N_2 = 11$. Calculating now the actual probabilities for evidence for u and v we get:*

$$\begin{aligned} P_{t,u}(E) &= 1-(1-\tfrac{1}{2}^{N_2})^{N_1} &&= 0.002 \\ P_{t,v}(E) &= 1-(\tfrac{1}{2}(1-\tfrac{1}{2}^{N_2}))^{N_1} &&= 0.984 \end{aligned}$$

6 Probabilistic Modal Logic and Probabilistic Bisimulation

We now address the strength of our test language in terms of the processes that may be distinguished by some testable property. The previous two sections show that the test language is at least strong enough to distinguish non–bisimular processes. However, even bisimular processes may be distinguished. As an example consider executing a test of the form $t = (a.b.\omega)^N$ on the processes p and q of figure 1 (which are clearly bisimular). In the resulting observation, $e = (e_1, \ldots, e_N)$, we expect the number of occurrences of $1_a : 1_b : 1_\omega$ as a component to be approximately $\frac{1}{2}N$ in the case of p, and $\frac{1}{3}N$ in the case of q. Also — a consequence of Chebyshev's Inequation (c.f. [Chu 74]) — the derivations from these expectations will decrease as N increases. Thus, it seems that we are indeed able to distinguish p and q by a test of the above form.

Of course, this should come as no surprise, as HML and the induced bisimulation equivalence only takes into consideration the *mere possibility* of transitions, and abstracts away from the actual *probability* with which a possible transition can take place. What is needed in order to characterize precisely the strength of our test language, seems to be probabilistic versions of HML and bisimulation. Thus, we propose below a Probabilistic Modal Logic (PML) with the $\langle a\rangle$ and $[a]$ modalities of HML being replaced by a continuum of modalities of the form $\langle a\rangle_\mu$, where a is an action and μ a probability.

Definition 6.1 *The formulas of PML are given by the following syntax:*

$$F ::= \mathrm{tt} \mid \mathrm{ff} \mid \Delta_a \mid F_1 \wedge F_2 \mid F_1 \vee F_2 \mid \langle a\rangle_\mu F$$

where $a \in Act$ and $\mu \in [0,1]$.

The *satisfaction relation*, $p \models F$, between processes and formulas of PML are defined as usual for $\mathrm{tt}, \mathrm{ff}, F_1 \wedge F_2$ and $F_1 \vee F_2$. $p \models \Delta_a$ holds whenever $p \stackrel{a}{\not\longrightarrow}$. Hence Δ_a corresponds to the formula $[a]\mathrm{ff}$ of HML. Now, we extend the $\longrightarrow$-notation in the following straightforward manner: for $S \subseteq Pr$ we write $p \stackrel{a}{\longrightarrow}_\mu S$, whenever $p \stackrel{a}{\longrightarrow}$ and $\sum_{q\in S} \mu_q = \mu$ where $\mu_q = \mu_{p,a}(q)$. Thus, we have as special cases, $p \stackrel{a}{\longrightarrow}_0 \emptyset$ and $p \stackrel{a}{\longrightarrow}_1 Pr$. Also, $p \stackrel{a}{\longrightarrow} S$ will abbreviate $p \stackrel{a}{\longrightarrow}_\mu S$ for some $\mu > 0$. Then, we define $p \models \langle a\rangle_\mu F$ whenever $p \stackrel{a}{\longrightarrow}_\nu S$ with $\nu \geq \mu$ and $\forall q \in S.q \models F$, for some S.

From this definition it should be clear that $p \models \langle a\rangle_{\frac{1}{2}}\langle b\rangle_1 \mathrm{tt}$ whereas $q \not\models \langle a\rangle_{\frac{1}{2}}\langle b\rangle_1 \mathrm{tt}$ (see figure 1).

Clearly, $\langle a\rangle_\nu F \Rightarrow \langle a\rangle_\mu F$, whenever $\nu \geq \mu$. Considering the modalities of HML, the following equivalences holds:

$$\begin{aligned} \langle a\rangle F &\equiv \exists \mu > 0.\langle a\rangle_\mu F \\ [a]F &\equiv \langle a\rangle_1 F \vee \Delta_a \end{aligned}$$

In case the minimal probability assumption holds, we may express $\langle a\rangle F$ directly as follows:

$$\langle a\rangle F \equiv \langle a\rangle_\epsilon F$$

Now, we face the problem of testing PML formulas, and in particular fomulas of the form $\langle a\rangle_\mu F$. Here, we only give an informal account: As for $\langle a\rangle$– and $[a]$–formulas of HML, the test for $\langle a\rangle_\mu F$ will have the form $(a.t_F)^N$, where t_F is a test for F, and N has to be determined by the desired level of significance. For $\langle a\rangle$– formulas respectively $[a]$–formulas the set of evidence was the tuples with *at least one* respectively *all* components confirming F. For the PML formula, $\langle a\rangle_\mu F$, the evidence set consists of all tuples, $(e_1, \ldots, e_N)$, where *at least* μN components are of the form $1_a : e_i'$ with e_i' confirming F. With this choice, the probabilities for evidence will — for increasing N — approach 0 respectively 1 for non–satisfying respectively satisfying processes (using Chebychev's Inequation [Chu 74]). Formalizing the above yields the following important result:

Theorem 6.2 *The formulas of Probabilistic Modal Logic are testable.*

To obtain an operational account of PML we refine the notion of bisimulation so that probabilities of transitions are catered for.

Whenever $\equiv$ is an equivalence on processes, we write $Pr/\equiv$ for the set of equivalence classes under $\equiv$. Using this notation, we may formulate the notion of bisimulation equivalence in the following alternative way:

$$\begin{aligned} &p \sim q \iff \\ &\quad \forall a \in Act. \forall S \in Pr/\sim . p \stackrel{a}{\longrightarrow} S \Leftrightarrow q \stackrel{a}{\longrightarrow} S \end{aligned}$$

Based on the formulation above we then obtain the notion of probabilistic bisimulation as follows:

Definition 6.3 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system. Then a* probabilistic bisimulation $\equiv$, *is an equivalence on Pr such that whenever $p \equiv q$, then the following holds:*

$$\forall a \in Act. \forall S \in Pr/\equiv .\ p \xrightarrow{a}_{\mu} S \Leftrightarrow q \xrightarrow{a}_{\mu} S$$

Two processes p and q are said to be probabilistic bisimular *in case (p, q) is contained in some probabilistic bisimulation. We write $p \equiv_P q$ in this case.*

Figure 4 shows two processes x and y, and a probabilistic bisimulation (or rather its equivalence classes) establishing $x \equiv_P y$.

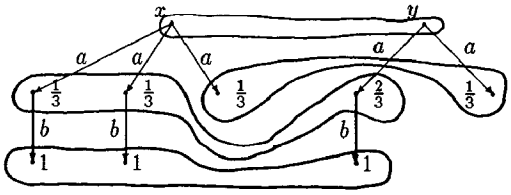

Figure 4. A probabilistic bisimulation.

Now, it may be proved that PML characterizes probabilistic bisimulation in the following sense:

Theorem 6.4 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system satisfying the minimal probability assumption. Then two processes are probabilistic bisimular just in case they satisfy exactly the same PML formulas.*

Even more importantly, it turns out that the notion of probabilistic bisimularity captures exactly the *limit* as to the distinguishing power of our test language: if two processes are probabilistic bisimular then *no* testable property will separate them. In fact, we can prove an even stronger claim: namely, that two processes are probabilistic bisimular just in case they yield the exact same probability distribution on the observation test of *any* test. We state this theorem without its (quite non-trivial) proof:

Theorem 6.5 *Let $\mathcal{P} = (Pr, Act, Can, \mu)$ be a probabilistic transition system satisfying the minimal probability assumption. Then $p \equiv_P q$ just in case $P_{t,p}(e) = P_{t,q}(e)$ for all tests t and observations $e \in \mathcal{O}_t$.*

7 Concluding Remarks and Future Work

We have presented a test language and a notion of (probabilistic) testability of process properties. In particular, we have demonstrated that properties expressed as formulas within Hennessy-Milner Logic are testable, and as a consequence that we may distinguish non-bisimular processes through testing. A Probabilistic Modal Logic has been introduced, and we have shown that the induced notion of probabilistic bisimularity characterizes the limit as to the distinguishing power of our test language.

The testability results are based on the assumptions of a minimal probability and a minimal deviation respectively. Intuitively the minimum probability assumption is necessary if a process property is going to be testable within a prescribed amount of time, while the minimum deviation assumption may seem a bit technical. However if one wants to test for a property af form $\langle a \rangle_\mu F$ without this assumption, the following adjustsments can be made:

- If it is essential that the applied test does not accept any process enjoying $\langle a \rangle_\nu F$, where ν is strictly less than μ, one can alternatively test for the property $\langle a \rangle_{\mu+\epsilon} F$ (ϵ beeing the minimal probability). Of course this has the drawback that processes enjoying $\langle a \rangle_\mu F$ may not pass the test.
- Another refinement is to choose a smaller minimal probability ε. This restricts the class of processes that may pass the test without enjoying property $\langle a \rangle_\mu F$, but it also increases the number of process copies that must be made.

An important issue for future studies is that of the *cost* of a test — e.g. measured by the number of basic experiments (pressure of buttons) required when executing the test. Obviously, when testing for a property with a given level of significance, we would prefer to use a test with lowest possible cost. Dually — a situation which might be more realistic — given an upper bound of the cost, what test will ensure the highest level of significance. Closely related to the cost of tests, is their *informativeness*. Loosely, we may consider a test t as being more informative than a test t', if any conclusion that can be derived from observations of t', also may be derived from observations of t. Obviously, we would expect more informative test, also to be more costly. An interesting problem for the future would be to axiomatize the information ordering on test.

Acknowledgements

This work — owing inspiration to the framework of Samson Abramsky [Abr 88] — has been carried out as part of the TAU-project [LarSkou 87], a project supported by the FTU-program under the danish research council. We would like to thank Susanne Christensen, Steffen Lauritzen, Søren Lundbye-Christensen and Aage Nielsen from the Statistics Group of our department for the many helpful discussions on the fundamentals of Statistics and Hypothesis Testing. Also thanks to Krik for making the pictures in this paper.

References

[Abr 88] S. Abramsky: *Observation Equivalence as a Testing Equivalence*, Theoretical Computer Science, 1988.

[BloIstrMey 88] B. Bloom, S. Istrail and A. Meyer: *Bisimulation Can't Be Traced: Preliminary Report*, Proceedings 15th ACM POPL, 1988.

[CoxHin 74] D.R. Cox and D.V. Hinkley: *Theoretical Statistics*, Chapman and Hall, 1974.

[Chu 74] K.L. Chung: *Elementary Probability Theory with Stochastic Processes*, Springer-Verlag, 1974.

[DeNicHen 84] R. De Nicola and M. Hennessy: *Testing Equivalences for Processes*, Theoretical Computer Science 34, 1984.

[HenMil 85] M. Hennessy and R. Milner: *Algebraic Laws for Nondeterminism and Concurrency*, Journal of the Association for computing Machinery, pp. 137–161, 1985.

[HughCres 68] G.E. Hughes and M.J. Cresswell: *An Introduction to Modal Logic*, Methuen, 1972.

[Koz 81] D. Kozen: *Semantics of Probabilistic Programs*, JCSS 22, 1981.

[Koz 83] D. Kozen: *A Probabilistic PDL*, Proceedings 10th ACM POPL, 1983.

[Lar 86] K.G. Larsen: *Context Dependent Bisimulation between Processes*, Ph.D Thesis, Edinburgh University, 1986.

[Lar 88] K.G. Larsen: *Proof Systems for Hennessy–Milner Logic with Recursion*, CAAP'88 Proceedings, LNCS 299.

[LarSkou 87] K.G. Larsen, A. Skou: *TAU: Theories for Concurrency, their Automation and Usage*, Aalborg University, Department of Mathematics and Computer Science, 1987.

[Mil 80] R. Milner: *Calculus of Communicating Systems*, LNCS 92.

[Mil 83] R. Milner: *Calculi for Synchrony and Asynchrony*, Theoretical Computer Science 25, 1983.

[Mil 81] R. Milner: *Modal Characterization of Observable Machine Behaviour*, CAAP'81 Proceedings, Lecture Notes in Computer Science 112.

[Phil 87] I. Phillips: *Refusal Testing* Theoretical Computer Science, 1987.

[Park 80] D. Park: *Concurrency and automata on infinite sequences*, Proc. 5th GI Conf., LNCS 104, 1981.

[Plotkin 81] G. Plotkin: *A Structural Approach to Operational Semantics*, Tech. Rep., DAIMI FN–19, Computer Sc., Aarhus University, Denmark, 1981.

[Pnue 85] A. Pnueli: *Linear and Branching Structures in the Semantics and Logics of Reactive Systems*, 12th ICALP, LNCS 194, 1985.

[PnuZuck 86] A. Pnueli, L. Zuck: *Verification of multiprocess probabilistic protocols*, Distributed Computing (1986) 1.

[Vardi 85] M.Y. Vardi: *Automatic verification of probabilistic concurrent finite–state programs*, Proceedings of 26th IEEE Symposium on Foundations of Computer Science, 1985.