Approximate Testing Equivalence Based on Time, Probability, and Observed Behavior

Alessandro Aldini
Institute of Information Science and Technology
University of Urbino, Italy
aldini@sti.uniurb.it

Several application domains require formal but flexible approaches to the comparison problem. Different process models that cannot be related by behavioral equivalences should be compared via a quantitative notion of similarity, which is usually achieved through approximation of some equivalence. While in the literature the classical equivalence subject to approximation is bisimulation, in this paper we propose a novel approach based on testing equivalence. As a step towards flexibility and usability, we study different relaxations taking into account orthogonal aspects of the process observations: execution time, event probability, and observed behavior. In this unifying framework, both interpretation of the measures and decidability of the verification algorithms are discussed.

1 Introduction

The need for a comparison between process models is an important requirement in several practical domains, ranging from the model-based verification of web service composition [16] to security [6], safety [18], and performability [1] verification. For instance, equivalence checking can be helpful to compare a web service implementation with some desired qualitative/quantitative service description, to relate an implemented software architecture to a reference dependable architectural model, and to reveal the performability impact of one component over the whole system through the comparison of the two system views that are obtained by activating/deactivating the component (this is generally called noninterference analysis). Such a comparison must be based on a precise semantics and some notion of process equivalence. In the formal methods community several notions of equivalence have been proposed which differ from each other for their observational power – e.g. from the "weakest" trace equivalence to bisimulation through the "intermediate" testing equivalence – and for their granularity – e.g. from nondeterministic versions of observation equivalences to the corresponding probabilistic, real-time, and stochastically timed extensions (see, e.g., [7] for a survey in the setting of process algebra).

In real-world applications perfect equivalence is usually hard to achieve when comparing models that may describe a system at different abstraction levels or different alternative implementations of the same ideal system. Hence, adding the quantitative aspect to the comparison is of paramount importance to establish how much these models fit according to an expected behavior. This can be done, e.g., in a framework where fine-grain models describe probability distributions of events or their temporal behaviors. Alternatively, functional models can be quantitatively compared with respect to a benchmark of testing scenarios. In any case, some kind of mathematical function must be employed to estimate the degree of similarity between processes that do not behave the same.

In this paper, a new approach to the approximation of behavioral equivalences is proposed in a process-algebraic setting in which three alternative dimensions – time, probability, and observed behavior – characterize what we mean by degree of similarity from the viewpoint of the expressive power of an external observer.

Submitted to:
QAPL 2010

First, we compare process models on the basis of their temporal behavior. Taking into account that time passes when observing the process execution requires the specification of durations. In our setting, system activities are associated directly with their durations, which are modeled through exponentially distributed random variables. In particular, the stochastic process governing the system evolution over time turns out to be a continuous-time Markov chain (CTMC). Second, by considering that timing aspects are dealt with probabilistically, it makes sense to compare process models with respect to probability distributions associated with their behaviors. Third, in order to analyze the observed behavior by abstracting from additional quantitative information such as time and probability, we introduce an approach that allows the distance between process models to be estimated with respect to their functional reaction to test-driven executions.

All the three dimensions are considered in the setting of a unifying semantics. More precisely, we employ a Markovian extension of testing equivalence [8], whose use represents a novelty in the field of approximate analysis. The main reason for this choice is that Markovian testing equivalence provides in a natural and explicit way an ideal framework for the definition of degree of similarity with respect to time, probability, and observed behavior. To give some intuitive insights, Markovian testing equivalence compares processes in terms of probability of observing test-driven computations that somehow "pass" tests and satisfy temporal constraints about the amount of time needed to pass these tests. Therefore, by relaxing in turn each of these parameters – durations associated with specific computations, probability distributions of these computations, and kind of tests elucidating them – we easily obtain different notions of approximate testing equivalence under the three considered dimensions. Moreover, as will be shown, in this framework it is possible to join the advantages of a decidable theory with the convenience of obtaining measures that can be easily interpreted in an activity oriented setting.

The remainder is organized as follows. First, we introduce some background about the testing framework (Sect. 2), i.e. we recall the Markovian process calculus and Markovian testing equivalence based on which we then formalize a notion of approximate testing equivalence from three different viewpoints (Sect. 3). The relaxed versions of Markovian testing equivalence based on time, probability, and observed behavior are presented separately and then combined in a unifying definition. Finally, the paper proposes some comparison with related work and interesting sights for future work (Sect. 4).

2 Markovian Testing Framework

In this section we recall Markovian testing equivalence in the setting of a Markovian process calculus that generates all the finite CTMCs with a minimum number of operators. For a complete survey of the main results concerning these topics, the interested reader is referred to [2].

2.1 Markovian Process Calculus

In the Markovian process calculus that we consider (MPC for short) every action is exponentially timed and its duration is described by a rate $\lambda \in \mathbb{R}_{>0}$ defining the exponential distribution such that the average duration of the action is given by the inverse of its rate. Formally, $Act = Name \times \mathbb{R}_{>0}$ is the set of actions of MPC, where *Name* is the set of action names, ranged over by $a, b, \ldots$, including the distinguished symbol τ denoting the invisible action.

The set of process terms of MPC is generated by the following syntax:

$$P ::= \underline{0} \mid <a,\lambda>.P \mid P + P \mid A$$

where:

- The inactive process $\underline{0}$ represents a terminated process.
- The action prefix operator $<a,\lambda>.P$ represents a process performing the durational action $<a,\lambda>$ and then behaving as P.
- The alternative composition operator encodes choice. If several durational actions can be performed the race policy is adopted, i.e. the fastest action is the one that is executed. The execution probability of each durational action is proportional to its rate and the average sojourn time associated with the related process term is exponentially distributed with rate given by the sum of the rates of the actions enabled by the term.
- A is a process constant defined by the possibly recursive equation $A \stackrel{\Delta}{=} P$.

We denote with $\mathscr{P}$ the set of closed and guarded process terms of MPC. The behavior of $P \in \mathscr{P}$ is given by the labeled multitransition system $[\![P]\!]$, where the states correspond to process terms and the transitions are labeled with actions. In particular, each transition has a multiplicity in order to keep track of the number of different proofs for the derivation of the transition. This is necessary because the idempotent law does not hold in the stochastic setting. Indeed, a term like $<a,\lambda>.P + <a,\lambda>.P$ is not the same as $<a,\lambda>.P$ because of the race policy.

From the labeled multitransition system $[\![P]\!]$ a CTMC can be easily derived by discarding the action names from the labels and collapsing all the transitions between any pair of states into a single transition whose rate is the sum of the rates of the collapsed transitions.

Formally, the semantic rules for MPC are as follows:

$$\langle a,\lambda\rangle.P \xrightarrow{a,\lambda} P$$

$$\frac{P_1 \xrightarrow{a,\lambda} P'}{P_1 + P_2 \xrightarrow{a,\lambda} P'} \qquad \frac{P_2 \xrightarrow{a,\lambda} P'}{P_1 + P_2 \xrightarrow{a,\lambda} P'}$$

$$\frac{A \stackrel{\Delta}{=} P \quad P \xrightarrow{a,\lambda} P'}{A \xrightarrow{a,\lambda} P'}$$

2.2 Markovian Testing Equivalence

Markovian testing equivalence is based on notions for process terms of MPC like exit rate – the rate at which we leave the state associated with the term – and computation – a sequence of transitions that can be executed starting from the state associated with the term. Below we recall these two notions before introducing the testing scenario.

Definition 2.1 Let $P \in \mathscr{P}$, $a \in \mathit{Name}$, and $C \subseteq \mathscr{P}$. The exit rate at which P executes actions of name a that lead to C is defined through the non-negative real function:

$$\mathit{rate}(P,a,C) = \sum\{\!|\, \lambda \in \mathbb{R}_{>0} \mid \exists P' \in C.\, P \xrightarrow{a,\lambda} P' \,|\!\}$$

where the summation is taken to be zero whenever its multiset is empty. ■

If we sum up the rates of all the actions that a process term P can execute, we obtain the total exit rate of P.

Definition 2.2 Let $P \in \mathscr{P}$. The total exit rate of P is defined as $rate_{\mathrm{t}}(P) = \sum_{a \in Name} rate(P, a, \mathscr{P})$. ■

The length of a computation is the number of transitions occurring in it. We denote with $\mathscr{C}_{\mathrm{f}}(P)$ the multiset of finite-length computations of $P \in \mathscr{P}$. Two distinct computations are independent of each other if neither is a proper prefix of the other one. In the remainder, we concentrate on finite sets of independent, finite-length computations. We now define the concrete trace, the probability, and the duration of an element of $\mathscr{C}_{\mathrm{f}}(P)$, using $\_ \circ \_$ for sequence concatenation and $|\_|$ for sequence length.

Definition 2.3 Let $P \in \mathscr{P}$ and $c \in \mathscr{C}_{\mathrm{f}}(P)$. The concrete trace associated with c is the sequence of action names labeling the transitions of c, which is defined by induction on the length of c through the $Name^*$-valued function:

$$trace(c) = \begin{cases} \delta & \text{if } |c| = 0 \\ a \circ trace(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c' \end{cases}$$

where δ is the empty trace. ■

Definition 2.4 Let $P \in \mathscr{P}$ and $c \in \mathscr{C}_{\mathrm{f}}(P)$. The probability of executing c is the product of the execution probabilities of the transitions of c, which is defined by induction on the length of c through the $\mathbb{R}_{]0,1]}$-valued function:

$$prob(c) = \begin{cases} 1 & \text{if } |c| = 0 \\ \frac{\lambda}{rate_{\mathrm{t}}(P)} \cdot prob(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c'. \end{cases}$$

We also define the probability of executing a computation in $C \subseteq \mathscr{C}_{\mathrm{f}}(P)$ as:

$$prob(C) = \sum_{c \in C} prob(c)$$

whenever C is finite and all of its computations are independent of each other. ■

Definition 2.5 Let $P \in \mathscr{P}$ and $c \in \mathscr{C}_{\mathrm{f}}(P)$. The stepwise average duration of c is the sequence of average sojourn times in the states traversed by c, which is defined by induction on the length of c through the $(\mathbb{R}_{>0})^*$-valued function:

$$time(c) = \begin{cases} \delta & \text{if } |c| = 0 \\ \frac{1}{rate_{\mathrm{t}}(P)} \circ time(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c' \end{cases}$$

where δ is the empty stepwise average duration. We also define the multiset of computations in $C \subseteq \mathscr{C}_{\mathrm{f}}(P)$ whose stepwise average duration is not greater than $\theta \in (\mathbb{R}_{>0})^*$ as:

$$C_{\leq \theta} = \{\!|\, c \in C \mid |c| \leq |\theta| \wedge \forall i = 1, \ldots, |c|.\, time(c)[i] \leq \theta[i] \,|\!\}.$$

Moreover, we denote by C^l the multiset of computations in $C \subseteq \mathscr{C}_{\mathrm{f}}(P)$ whose length is equal to $l \in \mathbb{N}$. ■

The main idea underlying the testing approach is that two process terms are equivalent whenever an external observer interacting with them by means of tests cannot infer any distinguishing information from the functional and quantitative standpoints. Tests are represented as process terms that interact with the terms to be tested through a parallel composition operator enforcing synchronization on all visible action names. A test is passed with success whenever a specific point during execution is reached. In the rest of the paper, we model tests as non-recursive, finite-state process terms.

Intuitively, at each state the process term proposes the execution of a durational action chosen according to the race policy and then, if such an action is visible, the test decides either to react by enabling the interaction or to block it (note that tests cannot block the execution of τ actions). The interaction can occur between actions with the same name only. If the test offers several actions with the same name as that of the action chosen by the term, then the selection of one such actions is probabilistic.

Formally, tests consist of nondurational actions each equipped with a weight $w \in \mathbb{R}_{>0}$. The set of tests respecting a canonical form is necessary and sufficient to decide whether two process terms are Markovian testing equivalent. Each of these canonical tests allows for one computation leading to success, whose intermediate states can have alternative computations leading to failure in one step.

Definition 2.6 The set $\mathbb{T}_{\mathrm{R,c}}$ of canonical reactive tests is generated by the syntax:

$$T ::= \mathrm{s} \mid <a,*_1>.T + \sum_{b \in \mathscr{E}-\{a\}} <b,*_1>.\mathrm{f}$$

where $a \in \mathscr{E}$, $\mathscr{E} \subseteq Name - \{\tau\}$ finite, the summation is absent whenever $\mathscr{E} = \{a\}$, and s (resp. f) is a zeroary operator standing for success (resp. failure). ■

The following semantic rules define the interaction between a process term and a test:

$$\frac{P \xrightarrow{\tau,\lambda} P'}{P \parallel T \xrightarrow{\tau,\lambda} P' \parallel T}$$

$$\frac{P \xrightarrow{a,\lambda} P' \qquad T \xrightarrow{a,*_w} T'}{P \parallel T \xrightarrow{a,\lambda \cdot \frac{w}{weight(T,a)}} P' \parallel T'}$$

where $weight(T,a) = \sum\{\!|\, w \mid \exists T'.\, T \xrightarrow{a,*_w}_{\mathrm{T}} T' \,|\!\}$ is the weight of T with respect to a and $\longrightarrow_{\mathrm{T}}$ denotes the transition relation for tests.

Given $P \in \mathscr{P}$ and $T \in \mathbb{T}_{\mathrm{R,c}}$, the interaction system of P and T is the process term $P \parallel T$, where each state of $[\![P \parallel T]\!]$ is called a configuration. We say that a configuration is successful if its test part is s and that a test-driven computation is successful if it traverses a successful configuration. We denote with $\mathscr{SC}(P,T)$ the multiset of successful computations of $P \parallel T$. It is worth noting that for any sequence $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time the multiset $\mathscr{SC}^{|\theta|}_{\leq \theta}(P,T)$ is finite and all the computations of it have a finite length and are independent of each other.

Markovian testing equivalence requires to compare the probabilities of performing successful test-driven computations within a given sequence of average amounts of time.

Definition 2.7 Let $P_1, P_2 \in \mathscr{P}$. We say that P_1 is Markovian testing equivalent to P_2, written $P_1 \sim_{\mathrm{MT}} P_2$, iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_2,T)).$$

■

Example 2.8 The following example justifies why the average duration of a computation has been defined in terms of the sequence of average sojourn times in the states traversed by the computation, rather than simply considering the sum of average durations. Take the two process terms:

$$<g,\gamma>.<a,\lambda>.<b,\mu>.\underline{0} + <g,\gamma>.<a,\mu>.<d,\lambda>.\underline{0}$$
$$<g,\gamma>.<a,\lambda>.<d,\mu>.\underline{0} + <g,\gamma>.<a,\mu>.<b,\lambda>.\underline{0}$$

Under the assumption $\lambda \neq \mu$ and $b \neq d$, both terms have a computation with concrete trace $g \circ a \circ b$, probability $\frac{1}{2}$, average duration $\frac{1}{2 \cdot \gamma} + \frac{1}{\lambda} + \frac{1}{\mu}$, but different average sojourn times. We can argue similarly for the computation with concrete trace $g \circ a \circ d$. Intuitively, an external observer distinguishes between them by observing the names of the actions that are performed and the instants at which they are performed. This is captured by $\sim_{\mathrm{MT}}$ as the two process terms are not Markovian testing equivalent. ■

3 Approximate Markovian Testing Equivalence

In this section we show three levels of approximation for $\sim_{\mathrm{MT}}$. The goal is to estimate from different perspectives how much a process term P_2 is similar to a given process term P_1. Here we assume that P_1 represents the original model to be approximated through an alternative model P_2 that must be compared with the original one. Since similarity cannot be transitive, as usual when relaxing equivalence relations we will also investigate what can be "transitively" inferred about the distance between two process terms P_1 and P_3 whenever there exists a process term P_2 that is similar to both of them.

The three considered dimensions of the similarity problem are: time taken to pass a test (Sect. 3.1), expressed as the sequence of average sojourn times in the states traversed by successful computations, probability with which tests are passed (Sect. 3.2), and syntactical form of the passed test (Sect. 3.3). For each dimension we will provide separately a measure of the distance between process terms that do not satisfy the $\sim_{\mathrm{MT}}$ relation, by stepwise refining the most adequate notion of similarity in terms of flexibility and usability. In each case, we will discuss the interpretation of the measure and the complexity of the algorithms measuring the distance between process terms. Finally, we will present a unifying notion of approximate Markovian testing equivalence – resulting in Def. 3.18 – which joins all the ingredients mentioned above. Indeed, a unifying framework is useful to study the trade-off existing among the three orthogonal aspects and the related impact upon the inequalities of the process terms under comparison. The proofs of results can be found in the appendix.

3.1 Approximating Time

The first dimension under consideration is time. In the setting of $\sim_{\mathrm{MT}}$, the time needed to pass a test with success is described as the sequence of average sojourn times in the states traversed by successful computations. Approximation at this level consists in relaxing the condition concerning the average sojourn times. We will introduce such an approximation through several steps in an incremental way. First, we will show how a process term P_1 can be approximated easily by a process term P_2 that is either "slightly slower" or "slightly faster" than P_1. For these interpretations, we first introduce the simplest relaxation of $\sim_{\mathrm{MT}}$ and then we show that it must be complicated in order to obtain a usable notion of similarity. Finally, we join both interpretations of similarity in order to obtain the most general definition of Markovian testing similarity with respect to time.

We start by introducing the idea of slow approximation. Whenever P_2 approximates successful computations of P_1 with respect to a test T and temporal threshold $\theta \in (\mathbb{R}_{>0})^*$, stepwise average sojourn times slightly greater than those imposed by θ may be tolerated. In this case we describe a slow approximation, in the sense that P_2 simulates P_1 – the same tests are passed with the same probabilities – but the successful computations of P_2 are slower than the corresponding ones of P_1.

As a first attempt in formalizing this intuition, we define the multiset of computations in $C \subseteq \mathscr{C}_{\mathrm{f}}(P)$ whose stepwise average duration is not greater than $\theta \in (\mathbb{R}_{>0})^*$ plus $\varepsilon \in \mathbb{R}_{\geq 0}$ as:

$$C_{\leq \theta + \varepsilon} = \{\!|\, c \in C \mid |c| \leq |\theta| \wedge \forall i = 1, \ldots, |c|.\, \mathit{time}(c)[i] \leq \theta[i] + \varepsilon \,|\!\}.$$

Based on this definition, we have the following relaxation of $\sim_{\mathrm{MT}}$.

Definition 3.1 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is slow Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon}(P_2,T)).$$ ■

Example 3.2 Consider the process terms $P_1 \stackrel{\Delta}{=} <g,\gamma>.<a,\gamma>.\underline{0}$ and $P_2 \stackrel{\Delta}{=} <g,\gamma-\delta>.<a,\gamma-\delta>.\underline{0}$. Then, P_2 approximates (is slow Markovian testing ε-similar to) P_1, where $\varepsilon = \frac{1}{(\gamma-\delta)} - \frac{1}{\gamma}$ expresses exactly the difference between the stepwise average amounts of time of the computations of P_1 and P_2. ■

Note that $P_1 \sim_{\mathrm{MT}} P_2$ if and only if P_2 (resp. P_1) is slow Markovian testing 0-similar to P_1 (resp. P_2). Moreover, we have the following transitivity result.

Proposition 3.3 Let $P_1, P_2, P_3 \in \mathscr{P}$ and $\varepsilon_1, \varepsilon_2 \in \mathbb{R}_{\geq 0}$. If P_2 is slow Markovian testing ε_1-similar to P_1 and P_3 is slow Markovian testing ε_2-similar to P_2, then P_3 is slow Markovian testing $(\varepsilon_1+\varepsilon_2)$-similar to P_1. ■

In favor of this approximation of $\sim_{\mathrm{MT}}$, we observe that it can be decided through a trivial variant of the algorithm for $\sim_{\mathrm{MT}}$ – which will be outlined later in this section – and with the same time complexity, which is $O(n^5)$, where n is the total number of states of $[\![P_1]\!]$ and $[\![P_2]\!]$ [2]. However, an approximation such as this is quite restrictive, as illustrated in the following example.

Example 3.4 Consider the process terms of the previous example. Then, P_2 is not slow Markovian testing ε'-similar to P_1, with $\varepsilon' > \frac{1}{(\gamma-\delta)} - \frac{1}{\gamma}$. In fact, take $\theta[1]$ such that $\theta[1] < \frac{1}{\gamma} \wedge \frac{1}{\gamma-\delta} < \theta[1] + \varepsilon'$. With this temporal threshold, any computation of P_1 is discarded, while this is not the case for P_2. ■

In order to further relax $\sim_{\mathrm{MT}}$, we need to compare explicitly the sets of computations of P_1 and P_2. Formally, given $C, C' \subseteq \mathscr{C}_{\mathrm{f}}(P)$, we now define the multiset of computations in C whose stepwise average duration is not greater than $\theta \in (\mathbb{R}_{>0})^*$ or else is ε-similar, with $\varepsilon \in \mathbb{R}_{\geq 0}$, to the stepwise average duration of any computation in $C'_{\leq\theta}$. Therefore:

$$\begin{aligned} C_{\leq\theta+\varepsilon,C'} = {} & C_{\leq\theta} \cup \\ & \{\!|\, c \in C \mid c \notin C_{\leq\theta} \wedge \exists c' \in C'_{\leq\theta}.\, |c| \leq |c'| \wedge \forall i = 1,\ldots,|c|.\, time(c')[i] \leq time(c)[i] \leq time(c')[i] + \varepsilon \,|\!\}. \end{aligned}$$

Based on this definition, we propose a new approximation of $\sim_{\mathrm{MT}}$.

Definition 3.5 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is slow Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon,\mathscr{SC}^{|\theta|}(P_1,T)}(P_2,T)).$$ ■

Intuitively, $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$ is compared with $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T)$ augmented with the successful T-driven computations of P_2 that are slower (up to ε) than corresponding computations in $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$.

Example 3.6 Consider two process terms P_1 and P_2 that are defined as follows, respectively:

$$<g,\gamma>.<a,\lambda>.<b,\lambda>.\underline{0} + <g,\gamma>.<a,\lambda>.<d,\lambda>.\underline{0}$$
$$<g,\gamma>.<a,\lambda>.<d,\lambda-\delta>.\underline{0} + <g,\gamma>.<a,\lambda-\delta>.<b,\lambda>.\underline{0}$$

The computation c_1 with concrete trace $g \circ a \circ b$ of P_1 is slowly ε-simulated by the corresponding computation c_2 of P_2, provided that $\varepsilon \geq \frac{1}{\lambda-\delta} - \frac{1}{\lambda}$. Given any test $T \in \mathbb{T}_{\mathrm{R,c}}$, for each $\theta \in (\mathbb{R}_{>0})^*$ we have that $c_1 \in \mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$ iff $c_2 \in \mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon,\mathscr{SC}^{|\theta|}(P_1,T)}(P_2,T)$, because from the temporal standpoint c_2 is stepwise slower than c_1 and their difference is limited by ε. We can argue similarly in the case of the two computations with concrete trace $g \circ a \circ d$. Hence, P_2 is slow Markovian testing ε-similar to P_1. ■

Note that $P_1 \sim_{\mathrm{MT}} P_2$ if and only if P_2 (resp. P_1) is slow Markovian testing 0-similar to P_1 (resp. P_2). Moreover, we have the following transitivity result.

Proposition 3.7 Let $P_1, P_2, P_3 \in \mathscr{P}$ and $\varepsilon_1, \varepsilon_2 \in \mathbb{R}_{\geq 0}$. If P_2 is slow Markovian testing ε_1-similar to P_1 and P_3 is slow Markovian testing ε_2-similar to P_2, then P_3 is slow Markovian testing δ-similar to P_1 for some $\delta \leq \varepsilon_1 + \varepsilon_2$. ■

Alternatively, by a symmetric argument a fast approximation can be defined whenever the successful computations of P_1 are approximated by successful computations of P_2 with stepwise average duration that is slightly lower than that of corresponding successful computations of P_1. Based on this intuition, we define the following approximation of $\sim_{\mathrm{MT}}$ still preserving the same results concerning Def. 3.5.

Definition 3.8 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is fast Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq \theta + \varepsilon, \mathscr{SC}^{|\theta|}(P_2,T)}(P_1, T)) = prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_2, T)).$$ ■

Example 3.9 Consider a variant of the previous example where the second process term is:

$$<g,\gamma>.<a,\lambda>.<d,\lambda+\delta>.\underline{0} + <g,\gamma>.<a,\lambda+\delta>.<b,\lambda>.\underline{0}$$

In this case it is easy to see that P_2 is fast Markovian testing ε-similar to P_1, where $\varepsilon \geq \frac{1}{\lambda} - \frac{1}{\lambda+\delta}$. ■

The definitions of (slow and fast) Markovian testing similarity can be decided in polynomial time by exploiting a simple variant of the same algorithm for $\sim_{\mathrm{MT}}$, because essentially the main objective – i.e. equating the execution probability of certain successful computations – does not change. The unique relaxation concerns the average durations of these computations, i.e. the criterion according to which the successful computations to compare are chosen. We now outline the most important steps of this proof by illustrating the differences with respect to the original algorithm for $\sim_{\mathrm{MT}}$ of [2]. First, deciding $\sim_{\mathrm{MT}}$ is reduced to decide the Markovian version of ready equivalence, which can be reduced to decide probabilistic ready equivalence if we consider the embedded discrete-time versions of the CTMCs underlying the two process terms to compare. Then, probabilistic ready equivalence is decided through a suitable reworking of the algorithm for probabilistic language equivalence [19]. In the transformation from continuous time to discrete time, information about the total exit rate of each state is encoded within the action names labeling the transitions leaving that state. Note that the use of this additional information provides the unique difference between $\sim_{\mathrm{MT}}$ and (slow and fast) Markovian testing similarity. More precisely, when applying the algorithm for probabilistic language equivalence in the case of $\sim_{\mathrm{MT}}$, a state of $[\![P_1]\!]$ is equated to a state of $[\![P_2]\!]$, i.e. they are put into the same accepting set, if and only if the two sets of augmented action names labeling the transitions departing from the two states coincide. In particular, they must exhibit the same total exit rates. Hence, the temporal information represents a decoration that is used to decide which states of $[\![P_1]\!]$ and $[\![P_2]\!]$ belong to the same accepting set. In our relaxed setting, instead of checking the equality between the total exit rates as required by $\sim_{\mathrm{MT}}$, we check their inequality up to ε, i.e. a state of $[\![P_1]\!]$ is equated to a state of $[\![P_2]\!]$ if the total exit rate of the second state is greater/lower than the total exit rate of the first state and their difference is limited by the threshold ε. Then, once the accepting sets are defined according to this condition, the algorithm of [19] proceeds as usual. The time complexity of the overall algorithm is $O(n^5)$.

Markovian testing similarity can be further relaxed. On the one hand, the fast and slow versions can be combined together, thus obtaining the following definition.

Definition 3.10 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is temporally Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq \theta + \varepsilon, \mathscr{SC}^{|\theta|}(P_2,T)}(P_1, T)) = prob(\mathscr{SC}^{|\theta|}_{\leq \theta + \varepsilon, \mathscr{SC}^{|\theta|}(P_1,T)}(P_2, T)).$$ ■

According to Def. 3.10, a computation c of P_1 can be approximated either by a slower or by a faster computation of P_2. However, c cannot be approximated by a computation of P_2 that is stepwise either slower or faster than c. To this aim, we introduce the following relaxation of $C_{\leq\theta+\varepsilon,C'}$:

$$C_{\leq\theta\pm\varepsilon,C'} = C_{\leq\theta} \cup \\ \{|\, c \in C \mid c \notin C_{\leq\theta} \wedge \exists c' \in C'_{\leq\theta}.\, |c| \leq |c'| \wedge \forall i = 1,\ldots,|c|.\, time(c')[i] - \varepsilon \leq time(c)[i] \leq time(c')[i] + \varepsilon \,|\}$$

Based on this notion of approximation, a computation c is similar to a computation c' if the difference between their average sojourn times is limited by ε. Then, we have the following variant of Def. 3.10.

Definition 3.11 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is temporally Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta\pm\varepsilon,\mathscr{SC}^{|\theta|}(P_2,T)}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta\pm\varepsilon,\mathscr{SC}^{|\theta|}(P_1,T)}(P_2,T)).$$ ■

In this way, a computation c of P_1 can be approximated by a computation of P_2 that is stepwise faster or slower than c. Such an extension does not alter the decidability results of Markovian testing similarity.

Example 3.12 Consider a variant of the previous example where the second process term is:

$$<g,\gamma>.<a,\lambda-\delta>.<d,\lambda+\delta>.\underline{0} + <g,\gamma>.<a,\lambda+\delta>.<b,\lambda-\delta>.\underline{0}$$

It can be verified that P_2 is temporally Markovian testing ε-similar to P_1, where $\varepsilon \geq \frac{1}{\lambda-\delta} - \frac{1}{\lambda}$. ■

On the other hand, when comparing the computations of two process terms we can decide to change at each step the value of the threshold expressing the tolerance to different temporal behaviors. This is obtained by assuming $\varepsilon \in (\mathbb{R}_{\geq 0})^*$ and checking the inequality:

$$\forall i = 1,\ldots,|c|.\, time(c')[i] \leq time(c)[i] \leq time(c')[i] + \varepsilon[i]$$

within the definition of $C_{\leq\theta+\varepsilon,C'}$. This variant can be used, e.g., to discount the effect of far (in the future) steps by assuming that $\varepsilon[i]$ increases as long as i increases.

3.2 Approximating Probability

The introduction of a relaxation concerning the probabilistic behavior of process terms results into the following extension of $\sim_{\mathrm{MT}}$ where the probabilities of the successful T-driven computations of P_1 and P_2 are not imposed to be equal anymore.

Definition 3.13 Let $P_1, P_2 \in \mathscr{P}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$. We say that P_2 is probabilistically Markovian testing ε-similar to P_1 iff for all reactive tests $T \in \mathbb{T}_{\mathrm{R,c}}$ and sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$|prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) - prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T))| \leq \varepsilon.$$ ■

As we have seen in the previous section, verifying Markovian testing equivalence amounts to decide whether two probabilistic automata accept the same words with the same probability. However, as shown in [10], the relaxation of this equivalence problem, i.e. checking whether for all words the distance between two process models is less than ε, is an undecidable problem.

To make it decidable, it is possible to restrict ourselves to more specific (and restrictive) notions of probabilistic similarity. As an example, [17] defines a polynomially accurate similarity that can be rephrased in our testing framework as follows: any set of successful computations of P_1 with a polynomial number of steps must be matched by P_2 with an error that is bounded by any polynomial. In order to measure the distance between process terms even when their difference is not negligible in the sense of [17], in the next section we will show that decidability is obtained by relaxing the condition over tests in Def. 3.13.

3.3 Approximating Tests

Similarly as done in Sect. 3.1, in this section we consider in an incremental way a notion of similarity that is based on the exemplary behavior of tests. The proposed approach is not completely naive as it is somehow inspired by [5], where processes are compared with respect to an event log describing typical behaviors. In particular, in [5] processes are defined in terms of Petri nets and an event log is a multiset of firing sequences. Then, different models are compared by measuring the overlap in (partially) fitting these sequences. This is done by using a fitness function and by taking into account all enabled transitions at any point in the sequence. This idea results into two measures, called precision and recall. Precision establishes whether the behavior of the second, alternative model is possible from the viewpoint of the behavior of the first, original model. Recall establishes how much of the behavior of the first model is covered by the second model. In our setting, we resort to a variant of this kind of approach from two different perspectives.

First, we observe that the notion of typical behavior that is at the base of model evaluation is naturally represented by tests. While in [5] it is suggested to define the event log through simulation or by explicitly describing by hand some typical behavior of interest, in our setting we formally describe an event log as a finite set of tests satisfying properties described in terms of logical formulas. Canonical tests do not exhibit any probabilistic and temporal behavior, so that for instance we can employ the logical characterization of testing equivalence, which comprises a restricted set of logical operators: a modal operator on sequences of visible actions, true, disjunction, and diamond [2]. Then, given a formula ϕ representing a property of interest, we use as event log the set of canonical tests satisfying ϕ, called $\mathbb{T}_{\mathrm{R,c},\phi}$, provided that such a set is finite. As an example, by following this idea ϕ could be the formula that is satisfied by all the tests in which the unique computation leading to success is made of the concrete trace $a_1 \circ \ldots \circ a_n$, representing the property with respect to which it is interesting to compare two process terms. In general, tests satisfying ϕ denote the set of typical behaviors parameterized by ϕ which guide the estimation of the degree of similarity between process terms.

Second, we observe that a test-based notion of the fitness measures of [5] can be used to estimate the similarity between tests. Approximating tests, as well as relaxing time and probability requirements, is justified by the fact that we intend to overcome the typical limitations of "perfect" equivalence. In order to relax $\sim_{\mathrm{MT}}$ by following this intuition, we assume that the process terms to compare should not exhibit the same quantitative behavior when interacting with the same test, but they can exhibit such a behavior when interacting with two possibly different but similar tests. In other words, if a process term satisfies a test with a certain probability and within a given amount of time, then the second one can simulate the behavior of the first term by satisfying with the same probability and by the same time another test that fits the first test according to a notion of test similarity.

Inspired by the formulas of [5], we now define the notions of behavioral precision and recall for test similarity. Let $trace_s(T)$ be the concrete trace associated with the unique computation of T leading to success, $|T|$ be the length of this trace, and T_i be the i-th process term of it, such that $T_1 ::= T$ and $T_{|T|}$ is the state that reaches success in one step. Then, we assume that $\forall i = 1,\ldots,|T|$, $enabled(T,i,s) = a$ iff $trace_s(T)[i] = a$ and $enabled(T,i,f) = \{b \mid T_i ::= <b,*_1>.\mathrm{f} + T'\}$. In practice, $enabled(T,i,s)$ denotes the transition belonging to the successful computation of T that is enabled at the i-th step, while $enabled(T,i,f)$ denotes the set of transitions leading to failure in one step that are enabled at the i-th step. Then, we introduce the following definitions of precision and recall for two tests T and T':

$$prec(T,T') \quad = \quad \frac{1}{|T'|} \sum_{i=1}^{|T'|} \frac{|(enabled(T,i,s) \cap enabled(T',i,s)) \cup (enabled(T,i,f) \cap enabled(T',i,f))|}{|enabled(T',i,f)| + |enabled(T',i,s)|}$$

and:

$prec(T_1,T_2)$	$rec(T_1,T_2)$	$prec(T_2,T_3)$	$rec(T_2,T_3)$	$prec(T_1,T_3)$	$rec(T_1,T_3)$
z	w	x	y	≤ 1	≤ 1
z	w	x	1	< 1	$\geq w$
z	w	1	y	≤ 1	$\leq w$
z	w	1	1	z	w
z	1	x	y	$\leq x$	≤ 1
z	1	x	1	$< x$	1
z	1	1	y	≤ 1	≤ 1
z	1	1	1	z	1
1	w	x	y	$\geq x$	≤ 1
1	w	x	1	$\geq x$	$\geq w$
1	w	1	y	1	$< w$
1	w	1	1	1	w
1	1	x	y	x	y
1	1	x	1	x	1
1	1	1	y	1	y
1	1	1	1	1	1

Table 1: Transitivity relations for *prec* and *rec*: $z,w,x,y \in [0,1[$

$$rec(T,T') \quad = \quad \frac{1}{|T|}\sum_{i=1}^{|T|} \frac{|(enabled(T,i,s) \cap enabled(T',i,s)) \cup (enabled(T,i,f) \cap enabled(T',i,f))|}{|enabled(T,i,f)| + |enabled(T,i,s)|}.$$

At each step we compare the set of enabled transitions for the current state of the two tests, by distinguishing the transitions leading to failure from the unique one along the computation leading to success. Both formulas establish a measure between 0 and 1 that estimates the similarity between them. Obviously, it holds that $prec(T,T') = rec(T',T)$. Similarly as in [5], it is important to note that tests are not imposed to offer the same behavior, which may differ step by step thus originating different computations.

Analogously, T and T' are not imposed to have the same length. For instance, if $|T| = 2 \cdot |T'| = 2 \cdot n$ and the behaviors of T and T' coincide in the first n steps, then $prec(T,T') = 1$ because each behavior of T' is possible according to the behavior of T, while $rec(T,T') = \frac{1}{2}$ because only half of the behavior of T is covered by the behavior of T'. On the other hand, T and T' coincide iff $prec(T,T') = rec(T,T') = 1$.

Example 3.14 Consider $T_1 = <a,*_1>.\mathrm{s} + <b,*_1>.\mathrm{f}$ and $T_2 = <b,*_1>.\mathrm{s} + <a,*_1>.\mathrm{f}$. Then, it holds that $prec(T_1,T_2) = rec(T_1,T_2) = 0$ because we distinguish actions leading to success from those leading to failure. Without this distinction, it would result $prec(T_1,T_2) = rec(T_1,T_2) = 1$.

Now, consider the two tests $T_1 = <a_1,*_1>.<a_2,*_1>.\mathrm{s} + <b,*_1>.\mathrm{f}$ and $T_2 = <c,*_1>.<a_2,*_1>.\mathrm{s} + <b,*_1>.\mathrm{f} + <b',*_1>.\mathrm{f}$. Then, $prec(T_1,T_2) = \frac{2}{3}$ and $rec(T_1,T_2) = \frac{3}{4}$. Recall is higher than precision, because the unique behavior of T_1 that is not covered by T_2 is the first action of the successful computation, while from the viewpoint of T_1 we have two impossible behaviors of T_2, i.e. the actions c and b'. ■

Precision and recall satisfy the same transitivity relations shown in [5], as reported in Table 1 for the sake of completeness (we omit the proofs, which are a recast of those in [5]).

Then, by using a notion of test similarity quantified with respect to the precision and recall defined above, we have the following relaxation of $\sim_{\mathrm{MT}}$, which is based on the observed behavior expressed in terms of test-driven computations, where instead of a single test we consider a pair of tests that fit almost the same. The first attempt abstracts from the temporal behavior of the process terms to compare.

Definition 3.15 Let $P_1, P_2 \in \mathscr{P}$ and $\mathbb{T}_{\mathrm{R,c},\phi}$ a finite set of tests. We say that P_2 is behaviorally Markovian testing similar to P_1 with precision $p \in [0,1]$ and recall $r \in [0,1]$ iff for each reactive test $T \in \mathbb{T}_{\mathrm{R,c},\phi}$ there exists a reactive test $T' \in \mathbb{T}_{\mathrm{R,c},\phi}$ such that:

1. $prec(T,T') \geq p$ and $rec(T,T') \geq r$
2. $prob(\mathscr{SC}(P_1,T)) = prob(\mathscr{SC}(P_2,T'))$. ■

As far as the transitivity properties of Def. 3.15 are concerned, we now discuss what can be inferred about two process terms P_1 and P_3 provided that there exists a process term P_2 such that P_2 is behaviorally Markovian testing similar to P_1 with precision p and recall r and P_3 is behaviorally Markovian testing similar to P_2 with precision p' and recall r'. By hypothesis, for each test T applied to P_1 there exists a test T' applied to P_2 such that the probabilities of the successful T-driven computations of P_1 and of the successful T'-driven computations of P_2 are equal. By hypothesis, there exists also a test T'' applied to P_3 such that the probabilities of the successful T'-driven computations of P_2 and of the successful T''-driven computations of P_3 are equal. Hence, the probabilities of the successful T-driven computations of P_1 and of the successful T''-driven computations of P_3 are equal. Afterwards, $prec(T,T'')$ and $rec(T,T'')$ can be inferred from p, r, p', and r' by following the conditions of Table 1.

In order to take into different account behaviors with a very low probability of success in comparison with successful behaviors occurring more frequent, in the two inequalities of Def. 3.15 we can multiply p and r by the probability of the successful test-driven computations of P_1.

The next step refines the condition about probabilities of Def. 3.15 by taking into account the temporal behavior of process terms. We recall that $\sim_{\mathrm{MT}}$ is defined with respect to all the sequences $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time. When considering a canonical test T and a process term P that does not execute invisible actions we can restrict ourselves to the sequences of length $|T|$, which is the exact number of steps needed to reach success. This is not enough to reduce the comparison between T and a similar test T' to a finite set of sequences. To this aim, we define a canonical set of sequences for T that is finite and is sufficient to decide whether a process term behaviorally simulates another one with respect to T.

Such a canonical set is made of a sequence for each subset of the set of successful computations $\mathscr{SC}^{|T|}(P,T)$. For each $X \in 2^{\mathscr{SC}^{|T|}(P,T)}$ we define the sequence of average amounts of time θ_X such that $\forall i = 1,\ldots,|T|.\, \theta_X[i] = \max_{c \in X}\{time(c)[i]\}$ and the canonical set $\Theta(P,T) = \{\theta_X \mid X \in 2^{\mathscr{SC}^{|T|}(P,T)}\}$. Note that $X \subseteq \mathscr{SC}^{|T|}_{\leq \theta_X}(P,T)$ and that we may have $\mathscr{SC}^{|T|}_{\leq \theta_X}(P,T) = \mathscr{SC}^{|T|}_{\leq \theta_Y}(P,T)$ for some $X \neq Y$, so that the minimum number of sequences to consider could be lower than $|2^{\mathscr{SC}^{|T|}(P,T)}|$.

The algorithm that computes these sequences consists of building a tree as follows. The root is at level 1 and is marked with the set of all the successful computations $\mathscr{SC}^{|T|}(P,T)$. If the current node of the level i is marked with a set $\mathscr{S}$ of computations, then create a child node for each $Y \subseteq \mathscr{S}$ for which there exists $k \in \mathbb{R}_{>0}$ such that $times(c)[i] \leq k$ for each $c \in Y$. Add to this new node the labels Y and $\max_{c \in Y}\{time(c)[i]\}$. The tree construction terminates at the level $|T|+1$. In this way, the tree contains at most $|2^{\mathscr{SC}^{|T|}(P,T)}|$ leafs, each leaf is associated with a subset $X \in 2^{\mathscr{SC}^{|T|}(P,T)}$, and the path from the root to this leaf contains as labels the average amounts of time forming the sequence θ_X.

Proposition 3.16 Let $P_1, P_2 \in \mathscr{P}$ and $T \in \mathbb{T}_{\mathrm{R,c}}$. If for each sequence $\theta \in \Theta(P_1,T) \cup \Theta(P_2,T)$ of average amounts of time we have:

$$prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_2,T))$$

then, we also have that for each sequence $\theta \in (\mathbb{R}_{>0})^*$ of average amounts of time:

$$prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq \theta}(P_2,T)).$$ ■

Now, we are ready to define a decidable approximation of $\sim_{\mathrm{MT}}$ based on observed behavior.

Definition 3.17 Let $P_1, P_2 \in \mathscr{P}$ and $\mathbb{T}_{\mathrm{R,c},\phi}$ a finite set of tests. We say that P_2 is behaviorally Markovian testing similar to P_1 with precision $p \in [0,1]$ and recall $r \in [0,1]$ iff for each reactive test $T \in \mathbb{T}_{\mathrm{R,c},\phi}$ there exists a reactive test $T' \in \mathbb{T}_{\mathrm{R,c},\phi}$ such that for all sequences $\theta \in \Theta(P_1,T) \cup \Theta(P_2,T')$ of average amounts of time:

1. $prec(T,T') \geq p$ and $rec(T,T') \geq r$
2. $prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T'))$. ■

The same considerations concerning the transitivity of Def. 3.15 still hold. With respect to the approximations based on time and probability that have been discussed in the previous sections, since in this setting we deal with finite sets of tests and sequences of average amounts of time it is possible to define a very intuitive, still decidable, approximation of $\sim_{\mathrm{MT}}$ based on time, probability, observed behavior, and the three corresponding families of quantitative thresholds.

Definition 3.18 Let $P_1, P_2 \in \mathscr{P}$ and $\mathbb{T}_{\mathrm{R,c},\phi}$ a finite set of tests. We say that P_2 is Markovian testing similar to P_1 with precision $p \in [0,1]$, recall $r \in [0,1]$, temporal threshold $\varepsilon \in \mathbb{R}_{>0}$, and probability threshold $\nu \in \mathbb{R}_{>0}$ iff for each reactive test $T \in \mathbb{T}_{\mathrm{R,c},\phi}$ there exists a reactive test $T' \in \mathbb{T}_{\mathrm{R,c},\phi}$ such that for all sequences $\theta \in \Theta(P_1,T) \cup \Theta(P_2,T')$ of average amounts of time:

1. $prec(T,T') \geq p$ and $rec(T,T') \geq r$
2. $|prob(\mathscr{SC}^{|\theta|}_{\leq\theta\pm\varepsilon,\mathscr{SC}^{|\theta|}(P_2,T')}(P_1,T)) - prob(\mathscr{SC}^{|\theta|}_{\leq\theta\pm\varepsilon,\mathscr{SC}^{|\theta|}(P_1,T)}(P_2,T'))| \leq \nu$. ■

Given a modal logic formula ϕ, we observe that P_2 (resp. P_1) is Markovian testing similar to P_1 (resp. P_2) with precision 1, recall 1, temporal and probability thresholds 0 if and only if $P_1 \sim_{\mathrm{MT}} P_2$ with respect to the tests defined by ϕ. It is worth noting that a unifying framework merging the three orthogonal aspects (time, probability, and observed behavior) puts the basis for the analysis of the trade-off among them.

Example 3.19 Consider two process terms P_1 and P_2 that are defined as follows, respectively:

$$<g,\gamma>.<a,\lambda+\delta>.<b,\lambda>.\underline{0} + <g,\gamma>.<a,\lambda>.<d,\lambda>.\underline{0}$$
$$<g,\gamma>.<a,\lambda>.<d',\lambda>.\underline{0} + <g,\gamma>.<a,\lambda>.<b,\lambda-\delta>.\underline{0}$$

and compare them with respect to tests whose successful computation is described by the concrete trace $g \circ a \circ *$, with $*$ any action. Then, P_2 is Markovian testing similar to P_1 with:

- both precision and recall equal to $\frac{2}{3}$, where the difference in the observed behaviors is due to the two concrete traces $g \circ a \circ d$ of P_1 and $g \circ a \circ d'$ of P_2, under the assumption $d \neq d'$;
- temporal threshold $\varepsilon \geq \frac{1}{\lambda-\delta} - \frac{1}{\lambda} > \frac{1}{\lambda} - \frac{1}{\lambda+\delta}$, where the difference in the average sojourn times is due to the three rates λ, $\lambda+\delta$, $\lambda-\delta$ labeling corresponding transitions related to the two concrete traces $g \circ a \circ b$ of P_1 and P_2;
- probability threshold 0, since the probabilities of the successful computations to compare are always the same. ■

4 Related and Future Work

In the last decade several approaches to the approximation of behavioral equivalences have been proposed, see, e.g., [15, 11, 20, 13, 4, 5, 12, 14] and the references therein.

For instance, some of them use a well-established approach based on behavioral pseudometrics [11, 20], which give a measure of the similarity between states of a transition system. These pseudometrics

provide a conservative extension of bisimulation equivalence. Hence, they cannot be compared with the notions of testing similarity, which instead rely on testing semantics. With approaches based on pseudometrics it is not easy to establish a clear relation between the measure estimating process similarity and its interpretation in a practical, mainly activity oriented, setting. As an example elucidating this aspect, [3] shows the importance of evaluating the impact that the absence of an equivalence relating two process terms has upon their difference in terms of performability properties – e.g. throughput measures – without, however, defining explicitly an approximate equivalence relating these measures with the degree of similarity. Some other approaches that are not based on pseudometrics, like [4, 12, 14], rely typically on relations approximating bisimulation equivalence. These approaches seem promising thanks to the strict relation between bisimulation and lumping for Markov chains [9]. Indeed, the characterization of lumpability is extremely useful, because the knowledge of a lumpable partition of the states of a Markov chain allows the generation of an aggregated Markov chain that is smaller than the original one, but leads to several results for the original Markov chain without an error. In this setting, there exist approximation techniques based on relaxed notions of lumping and on perturbation theory which establish bounds on the error made when approximating. This is particularly useful because these bounds are in direct relation with the numerical analysis of Markov chains and, therefore, provide immediately a clear interpretation of their impact upon the quantitative behavior of the process terms under analysis. However, it seems that there still exists a significant gap between the applicability of the approximate bisimulations mentioned above and their decidability. Very often, the (strict) assumptions underlying approximate bisimulation that are needed to define efficient verification algorithms are such that it becomes hard to find real application domains and, in particular, give a natural interpretation of the degree of similarity. On the other hand, the definition of an approximate bisimulation that can be related to approximate lumping and has an efficient verification algorithm is still an open problem.

Contrariwise, the approach proposed in [5] does not rely on behavioral equivalences, since it is based on the estimation of observed behaviors – quantified through a notion of fitness that does not require any nonfunctional information such as time and probability – whenever log-driven computations are compared. However, this estimation is not related to any notion of behavioral equivalence.

The main result of this paper is showing that testing equivalence offers an ideal semantic framework for joining ideas taken from approximate behavioral equivalences with those proposed in [5]. In addition, the proposed definitions of approximation elucidate the role of each aspect under consideration – time, probability, and observed behavior – without sacrificing in most cases neither decidability nor usability.

As future work, it would be interesting to investigate the relation between the estimations provided by approximate Markovian testing equivalence and T-lumpability [2], which is the version of lumpability corresponding to Markovian testing equivalence. One such result would enhance the applicability to domains where the degree of similarity must be interpreted in terms of impact upon the performance behavior of systems.

The application to real examples will be the subject of further investigations. For instance, it is well-known that approximate equivalence checking can be profitably employed in the setting of noninterference analysis. Basically, one user/component may affect the behavior of other users/components in a way that compromises properties like security and safety. Such an impact is studied by comparing the two views of the system interacting with these users/components that are obtained by activating and deactivating, respectively, the behavior of the interfering user/component. This approach is illustrated and used in [2] for the evaluation of performability aspects of several real-world case studies, like a secure routing system and a power-manageable system. In this setting, Markovian testing similarity could be employed to compare different system views with respect to families of properties formalized through modal logic formulas. The comparison would be conducted by distinguishing which observable

behaviors make these views different from functional, temporal, and probabilistic perspectives, each one accompanied by a measure of such a difference.

Acknowledgement
The author thanks the anonymous referees for their valuable comments. This work has been funded by MIUR-PRIN project *PaCo – Performability-Aware Computing: Logics, Models, and Languages*.

References

[1] A. Acquaviva, A. Aldini, M. Bernardo, A. Bogliolo, E. Bontà, and E. Lattanzi (2005): *A Formal Method Based Methodology for Predicting the Impact of Dynamic Power Management*. Formal Methods for Mobile Computing, Springer LNCS 3465, pp. 155–189.

[2] A. Aldini, M. Bernardo, and F. Corradini (2010): *A Process Algebraic Approach to Software Architecture Design*. Springer.

[3] A. Aldini and M. Bernardo (2009): *Weak Behavioral Equivalences for Verifying Secure and Performance-Aware Component-Based Systems*. Architecting Dependable Systems 6, Springer LNCS 5835, pp. 228–254.

[4] A. Aldini and A. Di Pierro (2008): *Estimating the Maximum Information Leakage*. Journal of Information Security 7, pp. 219–242.

[5] A.K. Alves de Medeiros, W.M.P. van der Aalst, and A.J.M.M. Weijters (2008): *Quantifying Process Equivalence Based on Observed Behavior*. Data & Knowledge Engineering 64, pp. 55–74.

[6] M. Backes, B. Pfitzmann, and M. Waidner (2007): *The Reactive Simulatability (RSIM) Framework for Asynchronous Systems*. Information and Computation 205, pp. 1685–1720.

[7] J.A. Bergstra, A. Ponse, and S.A. Smolka, Eds. (2001): *Handbook of Process Algebra*. Elsevier.

[8] M. Bernardo (2007): *Non-Bisimulation-Based Markovian Behavioral Equivalences*. Journal of Logic and Algebraic Programming 72, pp. 3–49.

[9] P. Buchholz (1994): *Exact and Ordinary Lumpability in Finite Markov Chains*. Journal of Applied Probability 31, pp. 59–75.

[10] M. de Rougemont and M. Tracol (2009): *Static Analysis for Probabilistic Processes*. Int. Symp. on Logic in Computer Science (LICS'09), IEEE-CS, pp. 299–308.

[11] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden (2004): *Metrics for Labelled Markov Processes*. Theoretical Computer Science 318, pp. 323–354.

[12] J. Desharnais, F. Laviolette, and M. Tracol (2008): *Approximate Analysis of Probabilistic Processes: Logic, Simulation and Games*. Int. Conf. on Quantitative Evaluation of Systems (QEST'08), IEEE-CS, pp. 264–273.

[13] A. Di Pierro, C. Hankin, and H. Wiklicky (2005): *Measuring the Confinement of Probabilistic Systems*. Theoretical Computer Science 340, pp. 3–56.

[14] A. Di Pierro, C. Hankin, and H. Wiklicky (2008): *Quantifying Timing Leaks and Cost Optimisation*. Conf. on Information and Comm. Security (ICICS'08), Springer LNCS 5308, pp. 81–96.

[15] P. Lincoln, J.C. Mitchell, M. Mitchell, and A. Scedrov (1999): *Probabilistic Polynomial-time Equivalence and Security Analysis*. World Congress on Formal Methods in the Development of Computing Systems (FM'99), Springer LNCS 1708, pp. 776–793.

[16] H. Foster, S. Uchitel, J. Magee, and J. Kramer (2003): *Model-based Verification of Web Service Compositions*. Int. Conf. on Automated Software Engineering (ASE'03), IEEE-CS, pp. 152–163.

[17] R. Segala and A. Turrini (2007): *Approximated Computationally Bounded Simulation Relations for Probabilistic Automata*. Computer Security Foundations Symposium (CSF'07), IEEE-CS, pp. 140–156.

[18] A. Simpson, J. Woodcock, and J. Davies (1998): *Safety through Security*. Workshop on Software Specification and Design (IWSSD'98), IEEE-CS pp. 18–24.

[19] W.G. Tzeng (1994): *A Polynomial-Time Algorithm for the Equivalence of Probabilistic Automata*. SIAM Journal on Computing 21, pp. 216–227.

[20] F. van Breugel and J. Worrell (2005): *A Behavioural Pseudometric for Probabilistic Transition Systems*. Theoretical Computer Science 331, pp. 115–142.

A Proof of Results

A.1 Proof of Prop. 3.3

Let $T \in \mathbb{T}_{\mathrm{R,c}}$ and $\theta \in (\mathbb{R}_{>0})^*$. By hypothesis, we have:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon_1}(P_2,T)).$$

Now consider θ' such that $|\theta'| = |\theta|$ and $\forall i = 1,\ldots,|\theta| : \theta'[i] = \theta[i] + \varepsilon_1$. Hence, by hypothesis we have:

$$prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'}(P_2,T)) = prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'+\varepsilon_2}(P_3,T)).$$

By definition of $C_{\leq\theta+\varepsilon}$ we immediately derive:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon_1+\varepsilon_2}(P_3,T)).$$ ■

A.2 Proof of Prop. 3.7

Let $T \in \mathbb{T}_{\mathrm{R,c}}$ and $\theta \in (\mathbb{R}_{>0})^*$. We distinguish between two cases.

On the one hand, let $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$ be empty. Hence, by hypothesis we have:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon_1,\mathscr{SC}^{|\theta|}(P_1,T)}(P_2,T)) = 0$$

from which, by definition of $C_{\leq\theta+\varepsilon,C'}$, it follows:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T)) = 0$$

and, again by hypothesis:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon_2,\mathscr{SC}^{|\theta|}(P_2,T)}(P_3,T)) = 0$$

from which the following result immediately follows:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\varepsilon_2,\mathscr{SC}^{|\theta|}(P_1,T)}(P_3,T)) = 0.$$

On the other hand, let $prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) > 0$. By hypothesis and by definition of $C_{\leq\theta+\varepsilon,C'}$, there exists θ' such that $|\theta'| = |\theta| \wedge \forall i = 1,\ldots,|\theta| :\ \theta'[i] = \theta[i] + \delta_1$, for some $\delta_1 \leq \varepsilon_1$, and:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)) = prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'}(P_2,T)).$$

Then, by hypothesis we also have:

$$prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'}(P_2,T)) = prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'+\varepsilon_2,\mathscr{SC}^{|\theta'|}(P_2,T)}(P_3,T)).$$

Analogously, there exists θ'' such that $|\theta''| = |\theta'| \wedge \forall i = 1,\ldots,|\theta'| :\ \theta''[i] = \theta'[i] + \delta_2$, for some $\delta_2 \leq \varepsilon_2$, and:

$$prob(\mathscr{SC}^{|\theta'|}_{\leq\theta'+\varepsilon_2,\mathscr{SC}^{|\theta'|}(P_2,T)}(P_3,T)) = prob(\mathscr{SC}^{|\theta''|}_{\leq\theta''}(P_3,T))$$

from which it follows:

$$prob(\mathscr{SC}^{|\theta''|}_{\leq\theta''}(P_3,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)).$$

Therefore, there must exist $\delta \leq \delta_1 + \delta_2$ such that:

$$prob(\mathscr{SC}^{|\theta|}_{\leq\theta+\delta,\mathscr{SC}^{|\theta|}(P_1,T)}(P_3,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)).$$ ■

A.3 Proof of Prop. 3.16

Let $\theta \in (\mathbb{R}_{>0})^*$. Since T is a canonical test we can just consider the prefix of θ of length $|T|$ (the case $|\theta| < |T|$ is trivial). We now distinguish two cases.

First, $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T) = \emptyset$. We now show that assuming $X = \mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T) \neq \emptyset$ leads to a contradiction. By hypothesis, we have $X = \mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)$, where $\theta_X \leq \theta$, i.e. $\forall i = 1\ldots|T|.\, \theta_X[i] \leq \theta[i]$. Hence, $prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T)) = prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)) = prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T))$ because $\theta_X \in \Theta(P_2,T)$. Since we have $\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T) \subseteq \mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$, we immediately obtain the contradiction.

Second, $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T) \neq \emptyset$. Let $X = \mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T)$. By construction, $\theta_X \leq \theta$ and $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T) = \mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T)$. Since $\theta_X \in \Theta(P_1,T)$ it holds $prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T)) = prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T))$. Towards a contradiction, assume $prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)) \neq prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T))$. Hence, there is $c \in \mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T)$ but $c \notin \mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)$, i.e. $\exists i \in \{1\ldots|T|\}.\, \theta_X[i] < time(c)[i] \leq \theta[i]$. Moreover, $Y = \mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T) = \mathscr{SC}^{|\theta_Y|}_{\leq\theta_Y}(P_1,T)$, with $\theta_X[i] < time(c)[i] \leq \theta_Y[i] \leq \theta[i]$ and, in general, $\theta_X < \theta_Y < \theta$. Since $\theta_Y \in \Theta(P_2,T)$ it holds $prob(\mathscr{SC}^{|\theta_Y|}_{\leq\theta_Y}(P_2,T)) = prob(\mathscr{SC}^{|\theta_Y|}_{\leq\theta_Y}(P_1,T)) \neq prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)) = prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T))$, from which we obtain $prob(\mathscr{SC}^{|\theta_Y|}_{\leq\theta_Y}(P_1,T)) \neq prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T))$ which contradicts $\mathscr{SC}^{|\theta|}_{\leq\theta}(P_1,T) = \mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_1,T)$. Therefore, it must be $prob(\mathscr{SC}^{|\theta_X|}_{\leq\theta_X}(P_2,T)) = prob(\mathscr{SC}^{|\theta|}_{\leq\theta}(P_2,T))$, from which we derive the result.

We can argue in the same way by exchanging the roles of P_1 and P_2 to obtain the proof. ■