Non-Bisimulation-Based Markovian Behavioral Equivalences

Marco Bernardo *

Università di Urbino "Carlo Bo"
Istituto di Scienze e Tecnologie dell'Informazione
Piazza della Repubblica 13, 61029 Urbino, Italy

Abstract

The behavioral equivalence that is typically used to relate Markovian process terms and to reduce their underlying state spaces is Markovian bisimilarity. One of the reasons is that Markovian bisimilarity is consistent with ordinary lumping. The latter is an aggregation for Markov chains that is exact, hence it guarantees the preservation of the performance characteristics across Markovian bisimilar process terms. In this paper we show that two non-bisimulation-based Markovian behavioral equivalences – Markovian testing equivalence and Markovian trace equivalence – induce at the Markov chain level an aggregation strictly coarser than ordinary lumping that is still exact. We then show that only Markovian testing equivalence may constitute a useful alternative to Markovian bisimilarity, as it turns out to be a congruence with respect to the typical process algebraic operators, while Markovian trace equivalence is not a congruence with respect to parallel composition.

Key words: process algebra, Markov chains, behavioral equivalences, bisimulation semantics, testing semantics, trace semantics.

1 Introduction

In order to account for performance aspects, in the last two decades algebraic process calculi [20,17,1,3] have been extended so that stochastic processes can be associated with their terms. In this field, the focus has primarily been on equipping process terms with performance models in the form of continuous-time Markov chains (CTMCs). Several Markovian process calculi have been proposed in the literature (see, e.g., [16,14,6] and the references

* Corresponding author: bernardo@sti.uniurb.it

therein). Although they differ for the action representation – durational actions vs. instantaneous actions separated from time passing – as well as for the synchronization discipline – asymmetric vs. symmetric – such Markovian process calculi share a common feature: Markovian bisimulation equivalence.

Markovian bisimilarity [16] is a semantic theory building on [20,19] that has proven to be useful to relate Markovian process terms and to reduce their underlying state spaces. Two Markovian process terms are bisimilar if they are able to mimic each other's behavior stepwise, both from the functional viewpoint and from the performance viewpoint. The reason of the success of Markovian bisimilarity is that – besides being decidable in polynomial time [14] and preserving branching-time properties [2] – it is appropriate both on the algebraic side and on the performance side. First, it is a congruence with respect to all the typical process algebraic operators [16], thus allowing for compositional reasoning and compositional state space reduction. Second, it has a sound and complete axiomatization [15], which elucidates the fundamental equational laws on which Markovian bisimilarity relies. Third, it is consistent with ordinary lumping [16], which means that Markovian bisimilarity makes sense from the performance viewpoint. Ordinary lumping [8] is an aggregation for Markov chains that is exact, i.e. the transient/stationary probability of being in a macrostate of an ordinarily lumped Markov chain is the sum of the transient/stationary probabilities of being in one of the constituent microstates of the original Markov chain. Thus, whenever two process terms are Markovian bisimilar, they are guaranteed to possess the same performance characteristics.

In the continuous-time setting research has mainly concentrated on branching-time equivalences [2] due to their connection with ordinary lumping. Only recently linear-time equivalences and testing scenarios have been investigated as well in the continuous-time case. In [7] Markovian testing equivalence has been proposed on the basis of [12,9,10]. Unlike Markovian bisimilarity, in which the ability to mimic the functional and performance behavior stepwise is taken into account, Markovian testing equivalence relies on a generic notion of efficiency measured by an external observer, which is given by the probability of passing tests within a certain amount of time. In [22] a behavioral equivalence for a process algebraic language based on probabilistic I/O automata has been considered, which is parameterized with respect to generic observables that associate real numbers with rated traces. In [23] the Markovian variants of several linear-time equivalences – trace equivalence, completed-trace equivalence, failure(-trace) equivalence, ready(-trace) equivalence [17,18] – have been investigated by means of push-button experiments conducted with appropriate Markovian trace machines.

All the Markovian behavioral equivalences defined in [7,22,23] are strictly coarser than Markovian bisimilarity, so at the CTMC level they result in aggregations that are strictly coarser than ordinary lumping. Although this can be helpful in practice to attack the state space explosion problem, we do not know whether such non-bisimulation-based Markovian behavioral equivalences make sense from the performance viewpoint. We are in fact facing the following open problem: are the CTMC-level aggregations induced by such equivalences exact? In other words, given two process terms that are related by one of these non-bisimulation-based Markovian behavioral equivalences, we do not know whether they possess the same performance characteristics.

The main contribution of this paper is to solve the above open problem by showing that both Markovian testing equivalence and Markovian trace equivalence induce at the CTMC level an aggregation strictly coarser than ordinary lumping that is still exact. This result ensures that any two process terms that are Markovian testing or trace equivalent possess the same performance characteristics. A further consequence is that Markovian testing and trace equivalences turn out to aggregate more than Markovian bisimilarity while preserving the exactness of the aggregation.

The strategy adopted in this paper to prove the exact aggregation property is to demonstrate first that Markovian testing and trace equivalences have sound and complete axiomatizations, which in turn requires to prove first that Markovian testing and trace equivalences are congruences. These two side results are provided for a basic Markovian process calculus with durational actions, which generates all the finite CTMCs with as few operators as possible: the null term, the action prefix operator, the alternative composition operator, and recursion. This ensures the general validity of the exact aggregation property without complicating the proof of the two side results. Once the axiomatizations of Markovian testing and trace equivalences have been obtained, we shall observe that they differ from the axiomatization of Markovian bisimilarity just for a new axiom schema subsuming one of the axioms of Markovian bisimilarity. As a consequence, in the proof of the exact aggregation property it will be necessary to concentrate only on the aggregations resulting from the application of this new axiom schema.

Besides the exact aggregation property and the congruence and axiomatization results proved on a sequential Markovian process calculus, this paper conveys further contributions related to Markovian testing and trace equivalences.

First, we exhibit several alternative characterizations for both equivalences. In particular, we show that considering the (more accurate) probability distributions quantifying the durations of the computations leads to the same two equivalences as considering the (easier to work with) average durations of the computations. For Markovian testing equivalence we additionally find a fully abstract characterization in terms of traces extended with the sets of the action names permitted at each step by the environment, which avoids analyzing the process term behavior in response to tests. It also turns out that such extended traces precisely characterize a set of canonical tests.

Second, we investigate the connections of Markovian testing and trace equivalences with their nondeterministic and probabilistic counterparts. We also study the connections among Markovian bisimilarity, Markovian testing equivalence, Markovian trace equivalence, and the completed, failure, ready, failure-trace and ready-trace variants of the latter, thus providing information about the Markovian linear-time/branching-time spectrum.

Third, we prove that Markovian testing equivalence is a congruence with respect to (a restricted version of) parallel composition, whereas Markovian trace equivalence is not. In light of both the exact aggregation property and this last result, of the two non-bisimulation-based Markovian behavioral equivalences considered in this paper, it turns out that only Markovian testing equivalence may constitute a useful alternative to Markovian bisimilarity.

This paper, which is a full and revised version of [7,4], is organized as follows. In Sect. 2 we introduce the syntax and the semantics for a sequential Markovian process calculus (SMPC) and a concurrent Markovian process calculus (CMPC), together with some notation that will be used several times throughout the rest of the paper. In Sect. 3 we recall the definition and the properties of Markovian bisimilarity over CMPC. In Sect. 4 we define Markovian testing equivalence over SMPC and we prove that it has a number of alternative characterizations, it has precise connections with other behavioral equivalences, it is a congruence with respect to the operators of SMPC, it has a sound and complete axiomatization over SMPC, and induces a CTMC-level aggregation coarser than ordinary lumping that is exact. In Sect. 5 we define Markovian trace equivalence over SMPC and we prove the same kind of properties enjoyed by Markovian testing equivalence; in particular, we show that both equivalences result in the same exact CTMC-level aggregation. In Sect. 6 we reconsider Markovian testing and trace equivalences over CMPC and we show that only Markovian testing equivalence is a congruence with respect to parallel composition. Finally, in Sect. 7 we report some concluding remarks.

2 Basic Markovian Process Calculi

In this section we introduce two basic Markovian process calculi with durational actions. The first one is a sequential Markovian process calculus (SMPC) and generates all the finite CTMCs with as few operators as possible: the null term, the action prefix operator, the alternative composition operator, and recursion. The second one is a concurrent Markovian process calculus (CMPC) as it additionally includes a parallel composition operator governed by an asymmetric synchronization discipline. Then we introduce some notation concerned with the exit rates of the process terms and the set of the attributes associated with their computations.

2.1 Syntax and Semantics for SMPC

In SMPC every action is durational, hence it is represented as a pair $<a, \lambda>$, where $a \in \mathit{Name}$ is the name of the action while $\lambda \in \mathbb{R}_{>0}$ is the rate of the exponential distribution quantifying the duration of the action. The average duration of an exponentially timed action is equal to the inverse of its rate.

Whenever several exponentially timed actions are enabled, the race policy is adopted, hence the fastest action is the one that is executed. As a consequence, in this generative [13] selection mechanism, the execution probability of any enabled exponentially timed action is proportional to its rate.

We denote by $\mathit{Act}_{\mathrm{S}} = \mathit{Name} \times \mathbb{R}_{>0}$ the set of the actions of SMPC. Unlike standard process theory, where a distinguished symbol τ is used as the name of the invisible action, here we assume that all the actions are observable.

Definition 2.1 The set of the process terms of SMPC is generated by the following syntax:

$$\boxed{\begin{array}{rcl} P & ::= & \underline{0} \\ & | & <a, \lambda>.P \\ & | & P + P \\ & | & A \end{array}}$$

where A is a process constant defined through the (possibly recursive) equation $A \triangleq P$. We denote by $\mathcal{P}_{\mathrm{S}}$ the set of the closed and guarded process terms of SMPC. ■

The semantics for SMPC can be defined in the usual operational style. As a consequence, the behavior of each process term is given by a multitransition system, whose states correspond to process terms and whose transitions – each of which has a multiplicity – are labeled with actions. From such a labeled multitransition system the CTMC underlying the process term can easily be retrieved by (i) discarding the action names from the transition labels and (ii) collapsing all the transitions between the same two states into a single transition whose rate is the sum of the rates of the original transitions.

We now provide the semantic rules for the various operators of SMPC:

- Null term: $\underline{0}$ cannot execute any action, hence the corresponding labeled multitransition system is just a state with no transitions.
- Exponentially timed action prefix: $<a, \lambda>.P$ can execute an action of name a and rate λ and then behaves as P:

$$\boxed{<a, \lambda>.P \xrightarrow{a,\lambda} P}$$

- Alternative composition: $P_1 + P_2$ behaves as either P_1 or P_2 depending on whether P_1 or P_2 executes an action first:

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

- Process constant: A behaves as the right-hand side process term in its defining equation:

$$\boxed{\frac{P \xrightarrow{a,\lambda} P' \quad A \triangleq P}{A \xrightarrow{a,\lambda} P'}}$$

2.2 Syntax and Semantics for CMPC

CMPC extends SMPC with a parallel composition operator governed by an asymmetric synchronization discipline, which is enforced on an explicit set of action names and makes use of passive actions. Multiway synchronizations are allowed provided that they involve at most one exponentially timed action, with all the other actions being passive.

A passive action is of the form $<a, *_w>$, where $w \in \mathbb{R}_{>0}$ is called weight and is used to quantify choices among passive actions with the same name. Every passive action has a duration that will become specified upon synchronization with an exponentially timed action having the same name.

Whenever several passive actions are enabled, the reactive [13] preselection policy is adopted. This means that, within every set of enabled passive actions with the same name, each such action is given an execution probability proportional to its weight. The choice between two enabled passive actions having different names is instead nondeterministic.

We denote by $Act_{\mathrm{C}} = Name \times Rate$ the set of the actions of CMPC, where $Rate = \mathbb{R}_{>0} \cup \{*_w \mid w \in \mathbb{R}_{>0}\}$ is the set of the action rates (ranged over by $\tilde{\lambda}$). As for SMPC, we assume that all the actions are observable.

Definition 2.2 The set of the process terms of CMPC is generated by the following syntax:

$$
\begin{array}{|l|}
\hline
P ::= \underline{0} \\
\quad\; | \;\; <a, \lambda>.P \\
\quad\; | \;\; <a, *_w>.P \\
\quad\; | \;\; P + P \\
\quad\; | \;\; P \parallel_S P \\
\quad\; | \;\; A \\
\hline
\end{array}
$$

where $S \subseteq \mathit{Name}$ and A is a process constant defined through the (possibly recursive) equation $A \triangleq P$. We denote by $\mathcal{P}_\mathrm{C}$ the set of the closed and guarded process terms of CMPC. ■

We now provide the semantic rules for the additional operators of CMPC:

- Passive action prefix: $<a, *_w>.P$ can execute a passive action of name a and weight w and then behaves as P:

$$
<a, *_w>.P \xrightarrow{a, *_w} P
$$

- Parallel composition: $P_1 \parallel_S P_2$ behaves as P_1 in parallel with P_2 as long as actions are executed whose names do not belong to S:

$$
\frac{P_1 \xrightarrow{a, \tilde{\lambda}} P_1' \quad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a, \tilde{\lambda}} P_1' \parallel_S P_2}
\qquad
\frac{P_2 \xrightarrow{a, \tilde{\lambda}} P_2' \quad a \notin S}{P_1 \parallel_S P_2 \xrightarrow{a, \tilde{\lambda}} P_1 \parallel_S P_2'}
$$

  with synchronizations being forced between any non-passive action executed by one term and any passive action executed by the other term that have the same name belonging to S:

$$
\frac{P_1 \xrightarrow{a, \lambda} P_1' \quad P_2 \xrightarrow{a, *_w} P_2' \quad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a, \lambda \cdot \frac{w}{weight(P_2, a)}} P_1' \parallel_S P_2'}
$$

$$
\frac{P_1 \xrightarrow{a, *_w} P_1' \quad P_2 \xrightarrow{a, \lambda} P_2' \quad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a, \lambda \cdot \frac{w}{weight(P_1, a)}} P_1' \parallel_S P_2'}
$$

  and between any two passive actions of the two terms that have the same name belonging to S:

$$\frac{P_1 \xrightarrow{a,*_{w_1}} P'_1 \quad P_2 \xrightarrow{a,*_{w_2}} P'_2 \quad a \in S}{P_1 \parallel_S P_2 \xrightarrow{a,*_{\frac{w_1}{weight(P_1,a)} \cdot \frac{w_2}{weight(P_2,a)} \cdot (weight(P_1,a)+weight(P_2,a))}} P'_1 \parallel_S P'_2}$$

where the weight of a process term P with respect to the passive actions of name a that it enables is defined as follows:

$$weight(P, a) = \sum\{\!| \, w \mid \exists P'. \, P \xrightarrow{a,*_w} P' \, |\!\}$$

We point out that the CTMC underlying a process term in $\mathcal{P}_{\mathrm{C}}$ can be retrieved only if its labeled multitransition system has no passive transitions. In this case we say that the process term is performance closed. We denote by $\mathcal{P}_{\mathrm{C,pc}}$ the set of the performance closed process terms of $\mathcal{P}_{\mathrm{C}}$. Note that $\mathcal{P}_{\mathrm{S,pc}} = \mathcal{P}_{\mathrm{S}}$.

2.3 Notation for Exit Rates and Computations

The Markovian behavioral equivalences that we shall define over SMPC and CMPC are based on concepts like the exit rates of the process terms and the traces, the probabilities, and the durations of their computations. Since these concepts will be used several times in the paper, we collect in this section the related notation.

By exit rate of a process term we mean the rate at which it is possible to leave the term. We distinguish among the rate at which the process term can execute actions of a given name that lead to a given set of terms, the total rate at which the process term can execute actions of a given name, and the total exit rate of the process term. The latter is the sum of the rates of all the actions that the process term can execute, and coincides with the reciprocal of the average sojourn time in the CTMC-level state corresponding to the process term whenever the process term is performance closed.

Since there are two kinds of actions – exponentially timed and passive – we consider a two-level definition of each variant of exit rate, where level 0 corresponds to exponentially timed actions and level -1 corresponds to passive actions.

Definition 2.3 Let $P \in \mathcal{P}_{\mathrm{C}}$, $a \in Name$, $l \in \{0, -1\}$, and $C \subseteq \mathcal{P}_{\mathrm{C}}$. The exit rate of P when executing actions of name a and level l that lead to C is defined through the following non-negative real function:

$$rate(P, a, l, C) = \begin{cases} \sum\{\!| \lambda \mid \exists P' \in C.\, P \xrightarrow{a,\lambda} P' |\!\} & \text{if } l = 0 \\ \sum\{\!| w \mid \exists P' \in C.\, P \xrightarrow{a,*_w} P' |\!\} & \text{if } l = -1 \end{cases}$$

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

Definition 2.4 Let $P \in \mathcal{P}_{\mathrm{C}}$ and $l \in \{0, -1\}$. The total exit rate of P at level l is defined through the following non-negative real function:

$$rate_{\mathrm{t}}(P, l) = \sum_{a \in Name} rate(P, a, l, \mathcal{P}_{\mathrm{C}})$$

where $rate(P, a, l, \mathcal{P}_{\mathrm{C}})$ is called the total exit rate of P with respect to a at level l. ■

By computation of a process term we mean a sequence of transitions that can be executed starting from the term. The length of a computation is given by the number of transitions occurring in it. We say that two computations are independent of each other if it is not the case that one of them is a proper prefix of the other one. In the following, we denote by $\mathcal{C}_{\mathrm{f}}(P)$ and $\mathcal{I}_{\mathrm{f}}(P)$ the multisets of the finite-length computations and independent computations of $P \in \mathcal{P}_{\mathrm{C}}$. Below we inductively define the trace, the execution probability, the average duration, and the duration distribution of an element of $\mathcal{C}_{\mathrm{f}}(P)$.

Definition 2.5 Let $P \in \mathcal{P}_{\mathrm{C}}$ and $c \in \mathcal{C}_{\mathrm{f}}(P)$. The trace associated with the execution of c is the sequence of the action names labeling the transitions of c, which is defined by induction on the length of c through the following $Name^*$-valued function:

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

where ε is the empty trace. ■

Definition 2.6 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$ and $c \in \mathcal{C}_{\mathrm{f}}(P)$. 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 following $\mathbb{R}_{]0,1]}$-valued function:

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

We also define the probability of executing a computation of C as:

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

for all $C \subseteq \mathcal{I}_{\mathrm{f}}(P)$. ■

Definition 2.7 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$ and $c \in \mathcal{C}_{\mathrm{f}}(P)$. The average duration of c is the sequence of the average sojourn times in the states traversed by c, which is defined by induction on the length of c through the following $\mathbb{R}^*_{>0}$-valued function:

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

where ε is the empty average duration. We also define the set of the computations of C whose average duration is not greater than θ as:

$$\boxed{\begin{aligned} C_{\leq\theta} = \{c \in C \mid\, & length(c) \leq length(\theta) \, \wedge \\ & \forall i = 1, \dots, length(c).\, time_{\mathrm{a}}(c)[i] \leq \theta[i]\} \end{aligned}}$$

for all $C \subseteq \mathcal{C}_{\mathrm{f}}(P)$ and $\theta \in \mathbb{R}^*_{>0}$. ■

Definition 2.8 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$, $C \subseteq \mathcal{I}_{\mathrm{f}}(P)$, and $\theta \in \mathbb{R}^*_{>0}$. The probability distribution of executing a computation of C within a sequence θ of time units on average is given by:

$$\boxed{prob_{\mathrm{a}}(C, \theta) = \sum_{c \in C}^{length(c) \leq length(\theta)} prob(c) \cdot \prod_{i=1}^{length(c)} \Pr(time_{\mathrm{a}}(c)[i] \leq \theta[i])}$$

where $\Pr(time_{\mathrm{a}}(c)[i] \leq \theta[i]) \in \{0, 1\}$ is the probability of the event "$time_{\mathrm{a}}(c)[i] \leq \theta[i]$". ■

Definition 2.9 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$ and $c \in \mathcal{C}_{\mathrm{f}}(P)$. The duration of c is the sequence of the random variables quantifying the sojourn times in the states traversed by c, which is defined by induction on the length of c through the following random-variable-sequence-valued function:

$$\boxed{time_{\mathrm{d}}(c) = \begin{cases} \varepsilon & \text{if } length(c) = 0 \\ X_{rate_{\mathrm{t}}(P,0)} \circ time_{\mathrm{d}}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c' \end{cases}}$$

where ε is the empty duration while $X_{rate_{\mathrm{t}}(P,0)}$ is the exponentially distributed random variable with rate $rate_{\mathrm{t}}(P, 0) \in \mathbb{R}_{>0}$. ■

Definition 2.10 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$, $C \subseteq \mathcal{I}_{\mathrm{f}}(P)$, and $\theta \in \mathbb{R}^*_{>0}$. The probability distribution of executing a computation of C within a sequence θ of time units is given by:

$$\boxed{prob_{\mathrm{d}}(C, \theta) = \sum_{c \in C}^{length(c) \leq length(\theta)} prob(c) \cdot \prod_{i=1}^{length(c)} \Pr(time_{\mathrm{d}}(c)[i] \leq \theta[i])}$$

where $\Pr(time_{\mathrm{d}}(c)[i] \leq \theta[i]) = 1 - \mathrm{e}^{-\theta[i]/time_{\mathrm{a}}(c)[i]}$ is the cumulative distribution function of the exponentially distributed random variable $time_{\mathrm{d}}(c)[i]$, whose expected value is $time_{\mathrm{a}}(c)[i]$. ■

We conclude by observing that the average duration (resp. duration) of a finite-length computation has been defined as the sequence of the average sojourn times (resp. of the random variables quantifying the sojourn times) in the states traversed by the computation. The same quantity could have been defined as the sum of the same basic ingredients, but this would not have been appropriate from the point of view of non-bisimulation-based Markovian behavioral equivalences.

Example 2.11 Consider the two following process terms:

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

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

with $\lambda \neq \mu$ and $b \neq d$. The first term has the two following maximal computations each with probability 1/2:

$$c_{1,1} \equiv \ . \xrightarrow{a,\gamma} . \xrightarrow{a,\lambda} . \xrightarrow{b,\mu} .$$

$$c_{1,2} \equiv \ . \xrightarrow{a,\gamma} . \xrightarrow{a,\mu} . \xrightarrow{d,\lambda} .$$

while the second term has the two following maximal computations each with probability 1/2:

$$c_{2,1} \equiv \ . \xrightarrow{a,\gamma} . \xrightarrow{a,\lambda} . \xrightarrow{d,\mu} .$$

$$c_{2,2} \equiv \ . \xrightarrow{a,\gamma} . \xrightarrow{a,\mu} . \xrightarrow{b,\lambda} .$$

If the average duration were defined as the sum of the average sojourn times, then $c_{1,1}$ and $c_{2,2}$ would have the same trace $a \circ a \circ b$ and the same average duration $\frac{1}{2 \cdot \gamma} + \frac{1}{\lambda} + \frac{1}{\mu}$, and similarly $c_{1,2}$ and $c_{2,1}$ would have the same trace $a \circ a \circ d$ and the same average duration $\frac{1}{2 \cdot \gamma} + \frac{1}{\mu} + \frac{1}{\lambda}$. This would lead to conclude that the two terms are equivalent, whereas an external observer equipped with a button-pushing machine displaying the names of the actions that are performed and the times at which they are performed [23] would distinguish between the two terms. ■

3 Markovian Bisimilarity for CMPC

The behavioral equivalence that is typically used to reason on the process terms of a calculus like CMPC is Markovian bisimilarity. In this section we recall from [16,15,6] its definition and its properties.

3.1 Equivalence Definition

The basic idea behind Markovian bisimilarity is to capture whether two process terms are able to mimic each other's functional and performance behavior stepwise. This is formalized through the comparison of the two process term exit rates when executing actions of the same name and of the same level that lead to the same class of terms.

Definition 3.1 An equivalence relation $\mathcal{B} \subseteq \mathcal{P}_{\rm C} \times \mathcal{P}_{\rm C}$ is a Markovian bisimulation iff, whenever $(P_1, P_2) \in \mathcal{B}$, then for all action names $a \in \mathit{Name}$, levels $l \in \{0, -1\}$, and equivalence classes $C \in \mathcal{P}_{\rm C}/\mathcal{B}$:

$$\mathit{rate}(P_1, a, l, C) = \mathit{rate}(P_2, a, l, C)$$

■

Since the union of all the Markovian bisimulations can be proved to be the largest Markovian bisimulation, the definition below follows.

Definition 3.2 Markovian bisimilarity, denoted by $\sim_{\rm MB}$, is the union of all the Markovian bisimulations. ■

3.2 Congruence Property

Markovian bisimilarity supports compositional reasoning and state space minimization, as it is a congruence with respect to all the operators of CMPC.

Theorem 3.3 Let $P_1, P_2 \in \mathcal{P}_{\rm C}$. Whenever $P_1 \sim_{\rm MB} P_2$, then:

(1) $<a, \tilde{\lambda}>.P_1 \sim_{\rm MB} <a, \tilde{\lambda}>.P_2$ for all $<a, \tilde{\lambda}> \in \mathit{Act}_{\rm C}$.
(2) $P_1 + P \sim_{\rm MB} P_2 + P$ and $P + P_1 \sim_{\rm MB} P + P_2$ for all $P \in \mathcal{P}_{\rm C}$.
(3) $P_1 \,\|_S\, P \sim_{\rm MB} P_2 \,\|_S\, P$ and $P \,\|_S\, P_1 \sim_{\rm MB} P \,\|_S\, P_2$ for all $S \subseteq \mathit{Name}$ and $P \in \mathcal{P}_{\rm C}$. ■

The basic equational laws characterizing Markovian bisimilarity over the set $\mathcal{P}_{\mathrm{C,nr}}$ of the non-recursive process terms of $\mathcal{P}_{\mathrm{C}}$ are elucidated by the set $\mathcal{A}^{\mathrm{MB}}$ of axioms shown in Table 1. As far as the expansion law $\mathcal{A}_6^{\mathrm{MB}}$ for parallel composition is concerned, I and J are finite index sets (if empty, the related summations are taken to be $\underline{0}$). The validity of this law is a consequence of the memoryless property of the exponentially distributed durations, which allows the semantics for the parallel composition operator to be defined in the usual interleaving style.

$$(\mathcal{A}_1^{\mathrm{MB}}) \qquad P_1 + P_2 = P_2 + P_1$$

$$(\mathcal{A}_2^{\mathrm{MB}}) \qquad (P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$$

$$(\mathcal{A}_3^{\mathrm{MB}}) \qquad P + \underline{0} = P$$

$$(\mathcal{A}_4^{\mathrm{MB}}) \qquad <a, \lambda_1>.P + <a, \lambda_2>.P = <a, \lambda_1 + \lambda_2>.P$$

$$(\mathcal{A}_5^{\mathrm{MB}}) \qquad <a, *_{w_1}>.P + <a, *_{w_2}>.P = <a, *_{w_1 + w_2}>.P$$

$$\begin{aligned}
(\mathcal{A}_6^{\mathrm{MB}}) \quad & \sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_{1,i} \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.P_{2,j} = \\
& \sum_{k \in I, a_k \notin S} <a_k, \tilde{\lambda}_k>. \left(P_{1,k} \parallel_S \sum_{j \in J} <b_j, \tilde{\mu}_j>.P_{2,j} \right) + \\
& \sum_{h \in J, b_h \notin S} <b_h, \tilde{\mu}_h>. \left(\sum_{i \in I} <a_i, \tilde{\lambda}_i>.P_{1,i} \parallel_S P_{2,h} \right) + \\
& \sum_{k \in I, a_k \in S, \tilde{\lambda}_k \in \mathbb{R}_{>0}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}} <a_k, \tilde{\lambda}_k \cdot \frac{w_h}{weight(P_2, b_h)}>.(P_{1,k} \parallel_S P_{2,h}) + \\
& \sum_{h \in J, b_h \in S, \tilde{\mu}_h \in \mathbb{R}_{>0}} \; \sum_{k \in I, a_k = b_h, \tilde{\lambda}_k = *_{v_k}} <b_h, \tilde{\mu}_h \cdot \frac{v_k}{weight(P_1, a_k)}>.(P_{1,k} \parallel_S P_{2,h}) + \\
& \sum_{k \in I, a_k \in S, \tilde{\lambda}_k = *_{v_k}} \; \sum_{h \in J, b_h = a_k, \tilde{\mu}_h = *_{w_h}} <a_k, \\
& \qquad *_{\frac{v_k}{weight(P_1, a_k)} \cdot \frac{w_h}{weight(P_2, b_h)} \cdot (weight(P_1, a_k) + weight(P_2, b_h))}>.(P_{1,k} \parallel_S P_{2,h})
\end{aligned}$$

Table 1
Axiomatization of $\sim_{\mathrm{MB}}$ over $\mathcal{P}_{\mathrm{C,nr}}$

Theorem 3.4 The deduction system $DED(\mathcal{A}^{\mathrm{MB}})$ is sound and complete for $\sim_{\mathrm{MB}}$ over $\mathcal{P}_{\mathrm{C,nr}}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\mathrm{C,nr}}$:

$$P_1 \sim_{\mathrm{MB}} P_2 \iff \mathcal{A}^{\mathrm{MB}} \vdash P_1 = P_2 \qquad \blacksquare$$

For the sake of simplicity, in Table 1 we have not considered the axioms dealing with recursion. We also observe that axioms $\mathcal{A}_1^{\mathrm{MB}}$-$\mathcal{A}_4^{\mathrm{MB}}$ result in a sound and complete deduction system for $\sim_{\mathrm{MB}}$ over $\mathcal{P}_{\mathrm{S,nr}}$.

Markovian bisimilarity is consistent with a notion of aggregation for CTMCs that is known under the name of ordinary lumping [8].

Definition 3.5 A partition $\mathcal{L}$ of the state space $\mathcal{S}$ of a CTMC is an ordinary lumping iff, whenever $s_1, s_2 \in L$ for some $L \in \mathcal{L}$, then for all equivalence classes $L' \in \mathcal{L}$:

$$\sum\{\!|\ \lambda \mid \exists s' \in L'.\ s_1 \xrightarrow{\lambda} s'\ |\!\} = \sum\{\!|\ \lambda \mid \exists s' \in L'.\ s_2 \xrightarrow{\lambda} s'\ |\!\}$$ ■

The fundamental property of an ordinary lumping of a CTMC is that the stochastic process resulting from it is a CTMC that is an exact aggregation of the original one. This means that the transient/stationary probability of being in a macrostate of the CTMC resulting from the ordinary lumping is the sum of the transient/stationary probabilities of being in one of the constituent microstates of the original CTMC.

Theorem 3.6 The CTMC-level aggregation induced by $\sim_{\rm MB}$ is an ordinary lumping. ■

Corollary 3.7 The CTMC-level aggregation induced by $\sim_{\rm MB}$ is exact. ■

An important consequence of this result is that, whenever two process terms of $\mathcal{P}_{\rm C,pc}$ are Markovian bisimilar, then they are guaranteed to possess the same performance characteristics. In other words, Markovian bisimilarity makes sense from the performance viewpoint, as it preserves the value of the performance measures across equivalent process terms.

To be more precise, this is the case unless we consider performance measures that distinguish between ordinarily lumpable states by assigning them different rewards. The interested reader is referred to [6] for a complete treatment of this issue.

4 Markovian Testing Equivalence for SMPC

In this section we introduce and study the properties of a non-bisimulation-based Markovian behavioral equivalence – originally defined in [7] – that captures a generic notion of efficiency measured by an external observer, which relies on the probability of passing tests within certain amounts of time. For the sake of simplicity, for the time being we present this Markovian testing equivalence by restricting ourselves to SMPC.

Two process terms can be considered equivalent if an external observer cannot distinguish between them. The only way that the observer has to infer information about the behavior of the two process terms is to interact with them by means of tests. In our Markovian framework with asymmetric communications, the most convenient way to represent a test is through another process term composed of passive actions only, which interacts with the terms to be tested by means of a parallel composition operator that enforces synchronization on any action name. In this way, the parallel composition of a performance closed term to be tested and a test will still be performance closed.

From the testing viewpoint, the idea is that in any of its states a process term to be tested generates the proposal of an action to be executed by means of a race among the exponentially timed actions enabled in that state. Then the test either reacts by participating in the interaction with the process term through a passive action having the same name as the proposed exponentially timed action, or blocks the interaction if it has no passive actions with the proposed name.

Since it is necessary to measure the probability with which process terms pass tests within a finite amount of time, for the test formalization we can restrict ourselves to non-recursive terms (composed of passive actions only). In other words, the expressiveness provided by labeled multitransition systems with a finite, dag-like structure will be enough for the tests.

In order to represent the fact that a test is passed or not, each of the terminal nodes of the dag-like semantic model underlying a test must be suitably labeled so as to establish whether it is a success or failure state. At the process calculus level, this amounts to replace $\underline{0}$ with the two zeroary operators "s" (for success) and "f" (for failure). Ambiguous terms like $\mathrm{s}+\mathrm{f}$ will be avoided in the test syntax by replacing the action prefix operator and the binary alternative composition operator with a set of n-ary guarded alternative composition operators, with n ranging over the whole $\mathbb{N}_{>0}$.

Definition 4.1 The set $\mathcal{T}$ of the tests is generated by the following syntax:

$$
\begin{array}{|lll|}
\hline
T & ::= & \mathrm{f} \\
 & | & \mathrm{s} \\
 & | & \sum\limits_{i \in I} <a_i, *_{w_i}>.T_i \\
\hline
\end{array}
$$

where I is a finite, non-empty index set. ■

Markovian testing equivalence relies on comparing the process term probabilities of performing a successful test-driven computation within a given average amount of time. A test-driven computation is a sequence of transitions in the labeled multitransition system underlying the parallel composition of a process term and a test. Due to the restrictions imposed on the tests in Sect. 4.1, all the considered test-driven computations will turn out to have a finite length, hence the inductive definitions of Sect. 2.3 apply to them.

Definition 4.2 Let $P \in \mathcal{P}_{\rm S}$ and $T \in \mathcal{T}$. The interaction system of P and T is process term $P \,\|_{Name}\, T \in \mathcal{P}_{\rm C,pc}$, where we say that:

- A configuration is a state of the labeled multitransition system underlying $P \,\|_{Name}\, T$.
- A configuration is successful (resp. failed) iff its test component is "s" (resp. "f").
- A computation is successful (resp. failed) iff so is its last configuration. A computation that is neither successful nor failed is said to be interrupted.

We denote by $\mathcal{SC}(P, T)$ the multiset of the successful computations of $\mathcal{C}_{\rm f}(P \,\|_{Name}\, T)$. ■

Note that $\mathcal{SC}(P, T) \subseteq \mathcal{I}_{\rm f}(P \,\|_{Name}\, T)$, because of the maximality of the successful test-driven computations, and that $\mathcal{SC}(P, T)$ is finite, because of the finitely-branching structure of the considered terms.

Definition 4.3 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. We say that P_1 is Markovian testing equivalent to P_2, written $P_1 \sim_{\rm MT} P_2$, iff for all tests $T \in \mathcal{T}$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

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

■

We now provide some necessary conditions for $\sim_{\rm MT}$, which are based on the average durations of successful test-driven computations exhibiting the same traces as well as on the total exit rates with respect to any action name of the process components of the last configurations of such computations. These necessary conditions will be useful in the remainder of Sect. 4. In the following, we denote by $\mathcal{C}_{\rm f}(T, {\rm s})$ the multiset of the (finite-length) computations of $T \in \mathcal{T}$ ending with s.

Definition 4.4 Let $P \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, and $c \in \mathcal{C}_{\rm f}(P \,\|_{Name}\, T)$. The projection of c on T is defined by induction on the length of c through the following

$\mathcal{C}_{\rm f}(T)$-valued function:

$$proj(c, T) = \begin{cases} T & \text{if } length(c) = 0 \\ T \xrightarrow{a, *_w} proj(c', T') & \text{if } c \equiv P \parallel_{Name} T \xrightarrow{a, \lambda} c' \\ & \quad \text{deriving from } T \xrightarrow{a, *_w} T' \end{cases}$$

where c' starts with $P' \parallel_{Name} T'$ for some a-derivative P' of P. ■

Definition 4.5 Let $P \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, $c \in \mathcal{C}_{\rm f}(T)$, and $c' \in \mathcal{C}_{\rm f}(P \parallel_{Name} T)$. We say that c' exercises c iff:

$$proj(c', T) = c$$

We denote by $\mathcal{EC}(P, T, c)$ and $\mathcal{ESC}(P, T, c)$ the multisets of the computations of $\mathcal{C}_{\rm f}(P \parallel_{Name} T)$ and $\mathcal{SC}(P, T)$ that exercise c. ■

Note that $\mathcal{EC}(P, T, c) \subseteq \mathcal{I}_{\rm f}(P \parallel_{Name} T)$ and $\mathcal{ESC}(P, T, c) \subseteq \mathcal{I}_{\rm f}(P \parallel_{Name} T)$ because the same test computation is exercised.

Proposition 4.6 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, and $c \in \mathcal{C}_{\rm f}(T, {\rm s})$. Whenever $P_1 \sim_{\rm MT} P_2$, then for all $c_k \in \mathcal{ESC}(P_k, T, c)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{ESC}(P_h, T, c)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$time_{\rm a}(c_k) = time_{\rm a}(c_h)$$

and for all $a \in Name$:

$$rate(P_{k,{\rm last}}, a, 0, \mathcal{P}_{\rm C}) = rate(P_{h,{\rm last}}, a, 0, \mathcal{P}_{\rm C})$$

with $P_{k,{\rm last}}$ (resp. $P_{h,{\rm last}}$) being the process component of the last configuration of c_k (resp. c_h).

Proof Taken $c_k \in \mathcal{ESC}(P_k, T, c)$, we proceed by induction on $length(c)$:

- If $length(c) = 0$ then necessarily $c \equiv T \equiv {\rm s}$. As a consequence, we immediately derive that there exists $c_h \in \mathcal{ESC}(P_h, T, c)$ such that:

  $$time_{\rm a}(c_k) = \varepsilon = time_{\rm a}(c_h)$$

  with $P_{k,{\rm last}} \equiv P_k \sim_{\rm MT} P_h \equiv P_{h,{\rm last}}$. Now suppose that $P_{k,{\rm last}}$ and $P_{h,{\rm last}}$ have a different total exit rate with respect to some $a \in Name$, say e.g.:

  $$rate(P_{k,{\rm last}}, a, 0, \mathcal{P}_{\rm C}) > rate(P_{h,{\rm last}}, a, 0, \mathcal{P}_{\rm C})$$

  If we considered the test $T' \equiv \, <a, *_1>.{\rm s}$, for $\theta = 1/rate(P_{k,{\rm last}}, a, 0, \mathcal{P}_{\rm C})$ we would have:

  $$prob(\mathcal{SC}_{\leq \theta}(P_{k,{\rm last}}, T')) = 1 \neq 0 = prob(\mathcal{SC}_{\leq \theta}(P_{h,{\rm last}}, T'))$$

  which contradicts $P_{k,{\rm last}} \sim_{\rm MT} P_{h,{\rm last}}$. As a consequence, it must be:

  $$rate(P_{k,{\rm last}}, a, 0, \mathcal{P}_{\rm C}) = rate(P_{h,{\rm last}}, a, 0, \mathcal{P}_{\rm C})$$

  for all $a \in Name$.

- Let $length(c) = n > 0$, with $c \equiv c'' \xrightarrow{b, *_w} \mathrm{s}$ and c'' ending with T''. Let T' be a test obtained from T by anticipating s by one step along every computation of T ending with s, and let c' be the consequent contraction by one step of c. Then $c' \in \mathcal{C}_{\mathrm{f}}(T', \mathrm{s})$. Let $c'_k \in \mathcal{ESC}(P_k, T', c')$ be the successful computation corresponding to the contraction by one step of c_k. Since $length(c') = n - 1$, by the induction hypothesis there exists $c'_h \in \mathcal{ESC}(P_h, T', c')$ such that:
$$time_{\mathrm{a}}(c'_k) = time_{\mathrm{a}}(c'_h)$$
and for all $a \in Name$:
$$rate(P'_{k,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}}) = rate(P'_{h,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}})$$
with $P'_{k,\mathrm{last}}$ (resp. $P'_{h,\mathrm{last}}$) being the process component of the last configuration of c'_k (resp. c'_h). As a consequence:
$$\begin{aligned} time_{\mathrm{a}}(c_k) &= time_{\mathrm{a}}(c'_k) \circ \frac{1}{rate_{\mathrm{t}}(P'_{k,\mathrm{last}} \,\|_{Name}\, T'', 0)} = \\ &= time_{\mathrm{a}}(c'_h) \circ \frac{1}{rate_{\mathrm{t}}(P'_{h,\mathrm{last}} \,\|_{Name}\, T'', 0)} = time_{\mathrm{a}}(c_h) \end{aligned}$$
where $c_h \in \mathcal{ESC}(P_h, T, c)$ is one of the successful computations corresponding to the extension by one step of c'_h according to c, which must exist in order not to violate $P_k \sim_{\mathrm{MT}} P_h$.
Now assume that for each such c_h there exists $a_h \in Name$ such that:
$$rate(P_{k,\mathrm{last}}, a_h, 0, \mathcal{P}_{\mathrm{C}}) \neq rate(P_{h,\mathrm{last}}, a_h, 0, \mathcal{P}_{\mathrm{C}})$$
where $P_{k,\mathrm{last}}$ (resp. $P_{h,\mathrm{last}}$) is the process component of the last configuration of c_k (resp. c_h). Then we can build a test that distinguishes P_k from P_h.
In fact, let us call strong sort a set of computations that intersects both $\mathcal{ESC}(P_k, T, c)$ and $\mathcal{ESC}(P_h, T, c)$ if it comprises all the computations with (the same trace and) the same average duration, whose last configurations have process components that enable actions with the same names and have the same total exit rate. Note that, due to $P_k \sim_{\mathrm{MT}} P_h$, the computations of a strong sort in $\mathcal{ESC}'(P_k, T, c)$ have the same probability as the computations of the strong sort in $\mathcal{ESC}'(P_h, T, c)$. We then say that a strong sort is rate-matching if for each computation of the strong sort in $\mathcal{ESC}(P_k, T, c)$ there exists a computation of the strong sort in $\mathcal{ESC}(P_h, T, c)$ such that the process components of their last configurations have the same total exit rates with respect to all action names, and vice versa. Note that, due to $P_k \sim_{\mathrm{MT}} P_h$, for all $a \in Name$ the probability of performing a computation of a rate-matching strong sort in $\mathcal{ESC}'(P_k, T, c)$ extended with an a-transition is the same as the probability of performing a computation of the rate-matching strong sort in $\mathcal{ESC}'(P_h, T, c)$ extended with an a-transition.
After removing from $\mathcal{ESC}(P_k, T, c)$ and $\mathcal{ESC}(P_h, T, c)$ every rate-matching strong sort, at least one of the two sets of remaining computations – which we denote by $\mathcal{ESC}'(P_k, T, c)$ and $\mathcal{ESC}'(P_h, T, c)$ – will be non-empty because of the assumption that c_k is not matched by any c_h deriving from the extension by one step of c'_h according to c. There are two cases.
First case: there exist some remaining computations in the same set, say e.g. $\mathcal{ESC}'(P_k, T, c)$, such that the process component of the last configuration of

each of them has the same maximum sum $\bar{r}$ of the total exit rates with respect to some $a_1, a_2, \ldots, a_m \in \mathit{Name}$ with $m \geq 1$, while the process component of the last configuration of each of the other remaining computations has a lower sum. In this case we build a test T''' deriving from c extended with a choice among m passive transitions labeled with $a_1, a_2, \ldots, a_m$ each leading to s.
Second case: the previous case does not apply. Then take the non-rate-matching strong sort such that the process component of the last configuration of each of its computations has the maximum total exit rate $\bar{r}$. Let $a_1, a_2, \ldots, a_m$ be the names of the actions enabled by each of these process components. In this case we build a test T''' deriving from c extended with a choice among m passive transitions labeled with $a_1, a_2, \ldots, a_m$, such that only some of them lead to s, while the others lead to f. Those leading to s have to be chosen on the basis of the different total exit rates with respect to $a_1, a_2, \ldots, a_m$ exhibited by the process components of the last configurations of the computations of the considered non-rate-matching strong sort, so that the extended versions of the computations of the sort in e.g. $\mathcal{ESC}'(P_k, T, c)$ get an higher probability than the extended versions of the computations of the sort in $\mathcal{ESC}'(P_h, T, c)$.
In each of the two cases, for some suitable $\bar{\theta}$ such that $\mathit{length}(\bar{\theta}) = \mathit{length}(c)$ we would have:

$$\mathit{prob}(\mathcal{SC}_{\leq \bar{\theta} \circ \frac{1}{\bar{r}}}(P_k, T''')) = p_k + q_k > p_h + q_h = \mathit{prob}(\mathcal{SC}_{\leq \bar{\theta} \circ \frac{1}{\bar{r}}}(P_h, T'''))$$

where $p_h = 0$ in the first case and $q_k, q_h \geq 0$ are the possible contributions of rate-matching strong sorts (whose average duration does not exceed $\bar{\theta}$ and whose last configurations have process components such that the sum of their total exit rates with respect to the names of the actions enabled at the end of T''' does not exceed $\bar{r}$), with $q_k = q_h$ by virtue of $P_k \sim_{\mathrm{MT}} P_h$. Since the above inequality contradicts $P_k \sim_{\mathrm{MT}} P_h$, for at least one c_h deriving from the extension by one step of c'_h according to c it must be:

$$\mathit{rate}(P_{k,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}}) = \mathit{rate}(P_{h,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}})$$

for all $a \in \mathit{Name}$. ■

Corollary 4.7 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$ and $T \in \mathcal{T}$. Whenever $P_1 \sim_{\mathrm{MT}} P_2$, then for all $c_k \in \mathcal{SC}(P_k, T)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{SC}(P_h, T)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$\mathit{trace}(c_k) = \mathit{trace}(c_h) \ \wedge \ \mathit{time}_{\mathrm{a}}(c_k) = \mathit{time}_{\mathrm{a}}(c_h)$$

■

Corollary 4.8 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Whenever $P_1 \sim_{\mathrm{MT}} P_2$, then for all $a \in \mathit{Name}$:

$$\mathit{rate}(P_1, a, 0, \mathcal{P}_{\mathrm{C}}) = \mathit{rate}(P_2, a, 0, \mathcal{P}_{\mathrm{C}})$$

■

Corollary 4.9 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Whenever $P_1 \sim_{\mathrm{MT}} P_2$, then:

$$\mathit{rate}_{\mathrm{t}}(P_1, 0) = \mathit{rate}_{\mathrm{t}}(P_2, 0)$$

■

4.3 Alternative Characterizations

We now provide the following alternative characterizations of Markovian testing equivalence:

(1) The first characterization is based on a predicate establishing whether a process term is able to pass a test with a probability that is above a given threshold, by taking an average time that is below another given threshold. This characterization, which is inspired by the may-pass and must-pass predicates that are at the basis of the classical definition of testing equivalence [12], will be useful in Sect. 4.4 to establish connections between $\sim_{\text{MT}}$ and its nondeterministic and probabilistic counterparts.
(2) The second characterization is based on the probability distribution of passing a test within a certain average amount of time. This characterization, which results in a slightly different definition with respect to the one of $\sim_{\text{MT}}$, will be useful to prove the third characterization.
(3) The third characterization is based on the probability distribution of passing a test within a certain amount of time. This characterization, which involves random variables instead of their expected values, justifies the definition of $\sim_{\text{MT}}$ in terms of the average durations of the test-driven computations, which are easier to work with than the probability distributions quantifying the same durations.
(4) The fourth characterization is based on traces that are suitably extended with the sets of the action names permitted at each step by the environment. This characterization – inspired by the one of [10] for probabilistic testing equivalence and consistent with the definition of the latter given in [9] – avoids analyzing the process term behavior in response to tests. Like in [10], a consequence of the structure of the proof will be the identification of a set of canonical tests, i.e. a set of tests that are necessary and sufficient in order to establish whether two process terms are Markovian testing equivalent. As we shall see, a canonical test admits a single computation leading to success, whose states can have additional computations each leading to failure in one step.
The fourth characterization will be useful in Sect. 5.3 to establish connections between $\sim_{\text{MT}}$ and Markovian trace equivalence and its variants, as well as in Sect. 6.1 to prove the congruence property of $\sim_{\text{MT}}$ with respect to parallel composition.

4.3.1 First Characterization: Pass Predicate

The first alternative characterization of $\sim_{\text{MT}}$ relies on a predicate establishing whether a process term can pass a test with a probability that is above a given threshold, by taking an average time that is below another given threshold.

Definition 4.10 Let $P \in \mathcal{P}_{\mathrm{S}}$, $T \in \mathcal{T}$, $p \in \mathbb{R}_{]0,1]}$, and $\theta \in \mathbb{R}^*_{>0}$. We say that P passes T with probability at least p within a sequence θ of time units on average, written $P \, \mathit{pass}_{p,\theta} \, T$, iff:

$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P,T)) \geq p \qquad \blacksquare$$

Definition 4.11 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian pass-testing equivalent to P_2, written $P_1 \sim_{\mathrm{MT,pass}} P_2$, iff for all tests $T \in \mathcal{T}$, probabilities $p \in \mathbb{R}_{]0,1]}$, and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

$$P_1 \, \mathit{pass}_{p,\theta} \, T \iff P_2 \, \mathit{pass}_{p,\theta} \, T \qquad \blacksquare$$

Proposition 4.12 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MT,pass}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2$$

Proof ($\Longrightarrow$) We prove the contrapositive. From $P_1 \not\sim_{\mathrm{MT}} P_2$ it follows that there exist $T \in \mathcal{T}$ and $\theta \in \mathbb{R}^*_{>0}$ such that e.g.:

$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_1,T)) = p > \mathit{prob}(\mathcal{SC}_{\leq\theta}(P_2,T))$$

Then $P_1 \, \mathit{pass}_{p,\theta} \, T$ whereas it is not the case that $P_2 \, \mathit{pass}_{p,\theta} \, T$. As a consequence $P_1 \not\sim_{\mathrm{MT,pass}} P_2$.
($\Longleftarrow$) Assume that $P_k \, \mathit{pass}_{p,\theta} \, T$ for some $k \in \{1,2\}$ and arbitrary $T \in \mathcal{T}$, $p \in \mathbb{R}_{]0,1]}$, and $\theta \in \mathbb{R}^*_{>0}$. Then $\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_k,T)) \geq p$. From $P_1 \sim_{\mathrm{MT}} P_2$ it follows that $\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_h,T)) \geq p$ for $h \in \{1,2\} - \{k\}$, hence $P_h \, \mathit{pass}_{p,\theta} \, T$ too. Therefore $P_1 \sim_{\mathrm{MT,pass}} P_2$. ■

4.3.2 Second Characterization: Average Durations

The second alternative characterization of $\sim_{\mathrm{MT}}$ is based on the probability distribution of passing a test within a certain average amount of time.

Definition 4.13 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian average-testing equivalent to P_2, written $P_1 \sim_{\mathrm{MT,a}} P_2$, iff for all tests $T \in \mathcal{T}$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

$$\mathit{prob}_{\mathrm{a}}(\mathcal{SC}(P_1,T),\theta) = \mathit{prob}_{\mathrm{a}}(\mathcal{SC}(P_2,T),\theta) \qquad \blacksquare$$

Lemma 4.14 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$, $C \subseteq \mathcal{I}_{\mathrm{f}}(P)$, and $\theta \in \mathbb{R}^*_{>0}$. Then:

$$\mathit{prob}_{\mathrm{a}}(C,\theta) = \mathit{prob}(C_{\leq\theta})$$

Proof It suffices to observe that for all $c \in C$ such that $\mathit{length}(c) \leq \mathit{length}(\theta)$:

$$\prod_{i=1}^{\mathit{length}(c)} \Pr(\mathit{time}_{\mathrm{a}}(c)[i] \leq \theta[i]) = \begin{cases} 1 & \text{if } c \in C_{\leq\theta} \\ 0 & \text{if } c \notin C_{\leq\theta} \end{cases} \qquad \blacksquare$$

Proposition 4.15 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MT,a}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2$$

Proof A straightforward consequence of Lemma 4.14. ■

4.3.3 Third Characterization: Duration Distributions

The third alternative characterization of $\sim_{\rm MT}$ is based on the probability distribution of passing a test within a certain amount of time. A consequence of this result is that considering the (more accurate) probability distributions quantifying the durations of the test-driven computations leads to the same equivalence as considering the (easier to work with) average durations of the test-driven computations, hence to $\sim_{\rm MT}$.

Definition 4.16 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. We say that P_1 is Markovian distribution-testing equivalent to P_2, written $P_1 \sim_{\rm MT,d} P_2$, iff for all tests $T \in \mathcal{T}$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of amounts of time:

$$prob_{\rm d}(\mathcal{SC}(P_1, T), \theta) = prob_{\rm d}(\mathcal{SC}(P_2, T), \theta)$$

■

Lemma 4.17 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, and $c \in \mathcal{C}_{\rm f}(T, {\rm s})$. Whenever $P_1 \sim_{\rm MT,d} P_2$, then for all $c_k \in \mathcal{ESC}(P_k, T, c)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{ESC}(P_h, T, c)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$time_{\rm d}(c_k) = time_{\rm d}(c_h)$$

and for all $a \in Name$:

$$rate(P_{k,\rm last}, a, 0, \mathcal{P}_{\rm C}) = rate(P_{h,\rm last}, a, 0, \mathcal{P}_{\rm C})$$

with $P_{k,\rm last}$ (resp. $P_{h,\rm last}$) being the process component of the last configuration of c_k (resp. c_h).

Proof Similar to the proof of Prop. 4.6. ■

Corollary 4.18 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$ and $T \in \mathcal{T}$. Whenever $P_1 \sim_{\rm MT,d} P_2$, then for all $c_k \in \mathcal{SC}(P_k, T)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{SC}(P_h, T)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$trace(c_k) = trace(c_h) \ \wedge \ time_{\rm d}(c_k) = time_{\rm d}(c_h)$$

■

Lemma 4.19 For $n \in \mathbb{N}_{>0}$ let:

- $\{p_i \in \mathbb{R}_{]0,1]} \mid 1 \leq i \leq n\}$ and $\{p'_i \in \mathbb{R}_{]0,1]} \mid 1 \leq i \leq n\}$ be such that $\sum_{i=1}^{n} p_i \leq 1$ and $\sum_{i=1}^{n} p'_i \leq 1$;
- $\{D_i : \mathbb{R}_{\geq 0} \longrightarrow \mathbb{R}_{[0,1]} \mid 1 \leq i \leq n\}$ and $\{D'_i : \mathbb{R}_{\geq 0} \longrightarrow \mathbb{R}_{[0,1]} \mid 1 \leq i \leq n\}$ be strictly increasing, non-linear, continuous functions such that $D_i(0) = 0 = D'_i(0)$ for each $i = 1, \ldots, n$, with $D_i \neq D_j$ and $D'_i \neq D'_j$ for $i \neq j$.

Whenever $D_i = D'_i$ for each $i = 1, \ldots, n$ and for all $t \in \mathbb{R}_{\geq 0}$:

$$\sum_{i=1}^{n} p_i \cdot D_i(t) = \sum_{i=1}^{n} p'_i \cdot D'_i(t)$$

then $p_i = p'_i$ for each $i = 1, \ldots, n$:

Proof Observed that for each $i = 1, \ldots, n$ it holds $D_i(t) = D'_i(t) = 0$ iff $t = 0$, let us consider the hypothesis rewritten for all $t \in \mathbb{R}_{>0}$ as follows:

$$\sum_{i=1}^{n} (p_i - p'_i) \cdot D_i(t) = 0$$

If we view each $p_i - p'_i$ as an unknown, say x_i, then we are facing a homogeneous linear system composed of uncountably many equations. Since the values belonging to the i-th column $(1 \leq i \leq n)$ of the coefficient matrix of the system are all positive and taken from the strictly increasing, non-linear function D_i, with such D_i's being all different from each other, the rows of the coefficient matrix are all linearly independent. As a consequence, the system admits only the solution $x_i = 0$, i.e. $p_i = p'_i$, for each $i = 1, \ldots, n$. ■

Theorem 4.20 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MT,d}} P_2 \iff P_1 \sim_{\mathrm{MT,a}} P_2$$

Proof Taken $T \in \mathcal{T}$, for each $k = 1, 2$ we denote by $\mathcal{S}^T_{k,l}$ the set of the computations of $\mathcal{SC}(P_k, T)$ with the same length $l \in \mathbb{N}$. Note that:

$$\mathcal{SC}(P_k, T) = \bigcup_{l \in \mathbb{N}} \mathcal{S}^T_{k,l}$$

In the following the computations of $\mathcal{S}^T_{k,l}$ with the same duration will be counted only once with their total probability. In order to exploit Lemma 4.19, we also assume that the computations of $\mathcal{S}^T_{k,l}$ with different durations are such that the products of the l elements of their average durations are all different, hence the products of the cumulative distribution functions of the l elements of their durations are all different. If this were not the case, without loss of generality we could consider two maximal subsets of $\mathcal{S}^T_{1,l}$ and $\mathcal{S}^T_{2,l}$ satisfying the above constraint, chosen in such a way that they respect both Cor. 4.18 and Cor. 4.7.
Since $\mathit{time}_{\mathrm{a}}(\_)[.]$ is the expected value of random variable $\mathit{time}_{\mathrm{d}}(\_)[.]$, given an arbitrary $\theta \in \mathbb{R}^*_{>0}$ such that $l \leq \mathit{length}(\theta)$ the result stems from the fact that the following equalities – in which functions *prob* and *time* are shortened by indicating only their initial – are all equivalent to each other:

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} \Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i]) = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} \Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])$$

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]} = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]}$$

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} \theta[i] \cdot \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]} = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} \theta[i] \cdot \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]}$$

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} \int_0^{\infty} \theta[i] \cdot \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]} \, d\theta[i] = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} \int_0^{\infty} \theta[i] \cdot \frac{d\Pr(t_{\mathrm{d}}(c)[i] \leq \theta[i])}{d\theta[i]} \, d\theta[i]$$

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} t_{\mathrm{a}}(c)[i] = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} t_{\mathrm{a}}(c)[i]$$

$$\sum_{c \in \mathcal{S}^T_{1,l}} p(c) \cdot \prod_{i=1}^{l} \Pr(t_{\mathrm{a}}(c)[i] \leq \theta[i]) = \sum_{c \in \mathcal{S}^T_{2,l}} p(c) \cdot \prod_{i=1}^{l} \Pr(t_{\mathrm{a}}(c)[i] \leq \theta[i])$$

In order to prove $\Longrightarrow$, we have initially exploited Cor. 4.18, the assumption about the products of the cumulative distribution functions of the l elements of the durations of the considered computations, and Lemma 4.19. In order to prove $\Longleftarrow$, we have initially exploited Cor. 4.7 via Prop. 4.15. Then the same conclusion as Lemma 4.19 has been reached. In fact, since $P_1 \sim_{\mathrm{MT,a}} P_2$ implies $P_1 \sim_{\mathrm{MT}} P_2$ by virtue of Prop. 4.15 and $\mathcal{S}^T_{1,l}$ and $\mathcal{S}^T_{2,l}$ collect all the successful computations with certain average durations, due to $P_1 \sim_{\mathrm{MT}} P_2$ for each such average duration the probability of performing a computation in $\mathcal{S}^T_{1,l}$ having that average duration must be equal to the probability of performing a computation in $\mathcal{S}^T_{2,l}$ having that average duration. Finally, in both directions we have exploited the fact that "=" is a congruence with respect to addition and multiplication, which has allowed us to substitute equals for equals within the same context "$\sum p(c) \cdot \prod$" of both sides of the equalities. ■

Corollary 4.21 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MT,d}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2$$

■

4.3.4 Fourth Characterization: Full Abstraction via Extended Traces

The fourth alternative characterization of $\sim_{\mathrm{MT}}$ is based on traces that are suitably extended with the sets of the action names permitted at each step by the environment. The meaning of this result is the possibility to characterize $\sim_{\mathrm{MT}}$ in a way that fully abstracts from the tests. The proof of this full abstraction result will consist of demonstrating that for each test computation there exists an extended trace with the same probabilistic and temporal characteristics as the test computation and that, conversely, for each extended trace there exists a test with the same probabilistic and temporal characteristics as the extended trace. A consequence of this proof structure will be the identification of a set of canonical tests, each having exactly one computation leading to success.

Definition 4.22 An element σ of $(\mathit{Name} \times 2^{\mathit{Name}})^*$ is an extended trace iff either σ is the empty sequence or:

$$\sigma \equiv (a_1, \mathcal{E}_1) \circ (a_2, \mathcal{E}_2) \circ \ldots \circ (a_n, \mathcal{E}_n)$$

for some $n \in \mathbb{N}_{>0}$ with $a_i \in \mathcal{E}_i$ for each $i = 1, \ldots, n$. We denote by $\mathcal{ET}$ the set of the extended traces. ■

Definition 4.23 Let $\sigma \in \mathcal{ET}$. The trace associated with σ is defined by induction on the length of σ through the following Name^*-valued function:

$$\mathit{trace}(\sigma) = \begin{cases} \varepsilon & \text{if } \mathit{length}(\sigma) = 0 \\ a \circ \mathit{trace}(\sigma') & \text{if } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

where ε is the empty trace. ■

Definition 4.24 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\sigma \in \mathcal{ET}$. We say that c is compatible with σ iff:

$$trace(c) = trace(\sigma)$$

We denote by $\mathcal{CC}(P, \sigma)$ the multiset of the computations of $\mathcal{C}_{\mathrm{f}}(P)$ that are compatible with σ. ■

Note that $\mathcal{CC}(P, \sigma) \subseteq \mathcal{I}_{\mathrm{f}}(P)$ because of the compatibility with the same σ.

Definition 4.25 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$, $\sigma \in \mathcal{ET}$, and $c \in \mathcal{CC}(P, \sigma)$. The probability of executing c with respect to σ is defined by induction on the length of c through the following $\mathbb{R}_{]0,1]}$-valued function:

$$prob^{\sigma}(c) = \begin{cases} 1 & \text{if } length(c) = 0 \\ \frac{\lambda}{\sum\limits_{b \in \mathcal{E}} rate(P, b, 0, \mathcal{P}_{\mathrm{C}})} \cdot prob^{\sigma'}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c' \\ & \quad \text{with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

We also define the probability of executing a computation of C with respect to σ as:

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

for all $C \subseteq \mathcal{CC}(P, \sigma)$. ■

Definition 4.26 Let $P \in \mathcal{P}_{\mathrm{C,pc}}$, $\sigma \in \mathcal{ET}$, and $c \in \mathcal{CC}(P, \sigma)$. The average duration of c with respect to σ is defined by induction on the length of c through the following $\mathbb{R}^{*}_{>0}$-valued function:

$$time^{\sigma}_{\mathrm{a}}(c) = \begin{cases} \varepsilon & \text{if } length(c) = 0 \\ \frac{1}{\sum\limits_{b \in \mathcal{E}} rate(P, b, 0, \mathcal{P}_{\mathrm{C}})} \circ time^{\sigma'}_{\mathrm{a}}(c') & \text{if } c \equiv P \xrightarrow{a,\lambda} c' \\ & \quad \text{with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

where ε is the empty average duration. We also define the set of the computations of C whose average duration with respect to σ is not greater than θ as:

$$C^{\sigma}_{\leq \theta} = \{c \in C \mid length(c) \leq length(\theta) \land \\ \forall i = 1, \dots, length(c).\ time^{\sigma}_{\mathrm{a}}(c)[i] \leq \theta[i]\}$$

for all $C \subseteq \mathcal{CC}(P, \sigma)$ and $\theta \in \mathbb{R}^{*}_{>0}$. ■

Definition 4.27 Let $P_1, P_2 \in \mathcal{P}_\mathrm{S}$. We say that P_1 is Markovian extended-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MTr,e}} P_2$, iff for all extended traces $\sigma \in \mathcal{ET}$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$prob^\sigma(\mathcal{CC}^\sigma_{\leq\theta}(P_1, \sigma)) = prob^\sigma(\mathcal{CC}^\sigma_{\leq\theta}(P_2, \sigma))$$

■

Definition 4.28 Let $T \in \mathcal{T}$ and $c \in \mathcal{C}_\mathrm{f}(T)$. The extended trace associated with the execution of c is defined by induction on the length of c through the following $\mathcal{ET}$-valued function:

$$trace_\mathrm{e}(c) = \begin{cases} \varepsilon & \text{if } length(c) = 0 \\ (a, \mathcal{E}) \circ trace_\mathrm{e}(c') & \text{if } c \equiv T \xrightarrow{a,*_w} c' \\ & \quad \text{with } \mathcal{E} = \{b \mid weight(T, b) > 0\} \end{cases}$$

where ε is the empty extended trace. ■

Definition 4.29 Let $T \in \mathcal{T}$ and $c \in \mathcal{C}_\mathrm{f}(T)$. The reactive probability of executing c is defined by induction on the length of c through the following $\mathbb{R}_{]0,1]}$-valued function:

$$prob_\mathrm{r}(c) = \begin{cases} 1 & \text{if } length(c) = 0 \\ \frac{w}{weight(T,a)} \cdot prob_\mathrm{r}(c') & \text{if } c \equiv T \xrightarrow{a,*_w} c' \end{cases}$$

where reactive is intended in the sense of [13]. ■

Lemma 4.30 Let $P \in \mathcal{P}_\mathrm{S}$, $T \in \mathcal{T}$, and $c \in \mathcal{C}_\mathrm{f}(T)$. Then for all $\theta \in \mathbb{R}_{>0}^*$:

$$prob(\mathcal{EC}_{\leq\theta}(P, T, c)) = prob_\mathrm{r}(c) \cdot prob^{trace_\mathrm{e}(c)}(\mathcal{CC}^{trace_\mathrm{e}(c)}_{\leq\theta}(P, trace_\mathrm{e}(c)))$$

Proof We proceed by induction on $length(c)$:

- If $length(c) = 0$ then $trace_\mathrm{e}(c) \equiv \varepsilon$, hence for all $\theta \in \mathbb{R}_{>0}^*$:
  $$prob(\mathcal{EC}_{\leq\theta}(P, T, c)) = 1 = prob_\mathrm{r}(c) \cdot prob^{trace_\mathrm{e}(c)}(\mathcal{CC}^{trace_\mathrm{e}(c)}_{\leq\theta}(P, trace_\mathrm{e}(c)))$$
- Let $length(c) = n > 0$, with $c \equiv T \xrightarrow{a,*_w} c'$ and T' being the first process term occurring in c'. We preliminarily observe that for some $r \in \mathbb{R}_{\geq 0}$:
  $$rate_\mathrm{t}(P \parallel_{Name} T, 0) = r = \sum_{b \in \mathcal{E}} rate(P, b, 0, \mathcal{P}_\mathrm{C})$$
  with $\mathcal{E} = \{b \mid weight(T, b) > 0\}$. Let $\theta \in \mathbb{R}_{>0}^*$. If $r = 0$ or $length(\theta) = 0$ or $\theta[1] < \frac{1}{r}$, then:
  $$prob(\mathcal{EC}_{\leq\theta}(P, T, c)) = 0 = prob_\mathrm{r}(c) \cdot prob^{trace_\mathrm{e}(c)}(\mathcal{CC}^{trace_\mathrm{e}(c)}_{\leq\theta}(P, trace_\mathrm{e}(c)))$$
  If instead $r > 0$ and $\theta = \frac{1}{\mu} \circ \theta'$ with $\frac{1}{\mu} \geq \frac{1}{r}$, then:

$$prob(\mathcal{EC}_{\leq\theta}(P,T,c)) = \sum_{P \xrightarrow{a,\lambda} P'} \frac{\lambda \cdot w / weight(T,a)}{r} \cdot prob(\mathcal{EC}_{\leq\theta'}(P',T',c'))$$

and:

$$prob^{trace_{\mathrm{e}}(c)}(\mathcal{CC}^{trace_{\mathrm{e}}(c)}_{\leq\theta}(P, trace_{\mathrm{e}}(c))) = \sum_{P \xrightarrow{a,\lambda} P'} \frac{\lambda}{r} \cdot prob^{trace_{\mathrm{e}}(c')}(\mathcal{CC}^{trace_{\mathrm{e}}(c')}_{\leq\theta'}(P', trace_{\mathrm{e}}(c')))$$

Since $length(c') = n - 1$, by the induction hypothesis we have:

$$prob(\mathcal{EC}_{\leq\theta'}(P',T',c')) = prob_{\mathrm{r}}(c') \cdot prob^{trace_{\mathrm{e}}(c')}(\mathcal{CC}^{trace_{\mathrm{e}}(c')}_{\leq\theta'}(P', trace_{\mathrm{e}}(c')))$$

hence the result. ■

Lemma 4.31 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Whenever $P_1 \sim_{\mathrm{MTr,e}} P_2$, then for all $T \in \mathcal{T}$, $c \in \mathcal{C}_{\mathrm{f}}(T)$, and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{EC}_{\leq\theta}(P_1,T,c)) = prob(\mathcal{EC}_{\leq\theta}(P_2,T,c))$$

Proof $P_1 \sim_{\mathrm{MTr,e}} P_2$ means that for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_1,\sigma)) = prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_2,\sigma))$$

hence in particular for all $T \in \mathcal{T}$, $c \in \mathcal{C}_{\mathrm{f}}(T)$, and $\theta \in \mathbb{R}^*_{>0}$:

$$prob^{trace_{\mathrm{e}}(c)}(\mathcal{CC}^{trace_{\mathrm{e}}(c)}_{\leq\theta}(P_1, trace_{\mathrm{e}}(c))) = prob^{trace_{\mathrm{e}}(c)}(\mathcal{CC}^{trace_{\mathrm{e}}(c)}_{\leq\theta}(P_2, trace_{\mathrm{e}}(c)))$$

The result then follows by virtue of Lemma 4.30 after multiplying both sides of the previous equality by $prob_{\mathrm{r}}(c)$. ■

Definition 4.32 Let $\sigma \in \mathcal{ET}$. The test associated with σ is defined by induction on the length of σ through the following $\mathcal{T}$-valued function:

$$test(\sigma) \triangleq \begin{cases} \mathrm{s} & \text{if } length(\sigma) = 0 \\ {<}a, *_1{>}.test(\sigma') + \sum_{b \in \mathcal{E} - \{a\}} {<}b, *_1{>}.\mathrm{f} & \text{if } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

where the summation is absent whenever $\mathcal{E} - \{a\} = \emptyset$. We denote by $\mathcal{T}_{\mathrm{et}}$ the set of the tests associated with the extended traces. ■

Lemma 4.33 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $\sigma \in \mathcal{ET}$. Then for all $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq\theta}(P, test(\sigma))) = prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P,\sigma))$$

Proof We proceed by induction on $length(\sigma)$:

- If $length(\sigma) = 0$ then $\sigma \equiv \varepsilon$ and $test(\sigma) \equiv \mathrm{s}$, hence for all $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq\theta}(P, test(\sigma))) = 1 = prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P,\sigma))$$

- Let $\mathit{length}(\sigma) = n > 0$, with $\sigma \equiv (a, \mathcal{E}) \circ \sigma'$. We preliminarily observe that for some $r \in \mathbb{R}_{\geq 0}$:
$$\mathit{rate}_{\mathrm{t}}(P \,\|_{\mathit{Name}}\, \mathit{test}(\sigma), 0) = r = \sum_{b \in \mathcal{E}} \mathit{rate}(P, b, 0, \mathcal{P}_{\mathrm{C}})$$
Let $\theta \in \mathbb{R}^*_{>0}$. If $r = 0$ or $\mathit{length}(\theta) = 0$ or $\theta[1] < \frac{1}{r}$, then:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P, \mathit{test}(\sigma))) = 0 = \mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P, \sigma))$$
If instead $r > 0$ and $\theta = \frac{1}{\mu} \circ \theta'$ with $\frac{1}{\mu} \geq \frac{1}{r}$, then:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P, \mathit{test}(\sigma))) = \sum_{P \xrightarrow{a,\lambda} P'} \frac{\lambda}{r} \cdot \mathit{prob}(\mathcal{SC}_{\leq\theta'}(P', \mathit{test}(\sigma')))$$
and:
$$\mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P, \sigma)) = \sum_{P \xrightarrow{a,\lambda} P'} \frac{\lambda}{r} \cdot \mathit{prob}^{\sigma'}(\mathcal{CC}^{\sigma'}_{\leq\theta'}(P', \sigma'))$$
Since $\mathit{length}(\sigma') = n - 1$, by the induction hypothesis we have:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta'}(P', \mathit{test}(\sigma'))) = \mathit{prob}^{\sigma'}(\mathcal{CC}^{\sigma'}_{\leq\theta'}(P', \sigma'))$$
hence the result. ■

Lemma 4.34 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then $P_1 \sim_{\mathrm{MTr,e}} P_2$ iff for all $T \in \mathcal{T}_{\mathrm{et}}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_1, T)) = \mathit{prob}(\mathcal{SC}_{\leq\theta}(P_2, T))$$

Proof $P_1 \sim_{\mathrm{MTr,e}} P_2$ means that for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_1, \sigma)) = \mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_2, \sigma))$$
By virtue of Lemma 4.33 this means that for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_1, \mathit{test}(\sigma))) = \mathit{prob}(\mathcal{SC}_{\leq\theta}(P_2, \mathit{test}(\sigma)))$$
■

Theorem 4.35 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:
$$P_1 \sim_{\mathrm{MTr,e}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2$$

Proof ($\Longrightarrow$) For each $k = 1, 2$ it holds that for all $T \in \mathcal{T}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_k, T)) = \sum_{c \in \mathcal{C}_{\mathrm{f}}(T, \mathrm{s})} \mathit{prob}(\mathcal{ESC}_{\leq\theta}(P_k, T, c))$$
Since $P_1 \sim_{\mathrm{MTr,e}} P_2$, from Lemma 4.31 it follows that for all $c \in \mathcal{C}_{\mathrm{f}}(T, \mathrm{s})$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{ESC}_{\leq\theta}(P_1, T, c)) = \mathit{prob}(\mathcal{ESC}_{\leq\theta}(P_2, T, c))$$
hence $P_1 \sim_{\mathrm{MT}} P_2$.
($\Longleftarrow$) If $P_1 \sim_{\mathrm{MT}} P_2$ then it holds in particular that for all $T \in \mathcal{T}_{\mathrm{et}}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_1, T)) = \mathit{prob}(\mathcal{SC}_{\leq\theta}(P_2, T))$$
hence $P_1 \sim_{\mathrm{MTr,e}} P_2$ by virtue of Lemma 4.34. ■

Corollary 4.36 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then $P_1 \sim_{\mathrm{MT}} P_2$ iff for all $T \in \mathcal{T}_{\mathrm{et}}$ and $\theta \in \mathbb{R}^*_{>0}$:
$$\mathit{prob}(\mathcal{SC}_{\leq\theta}(P_1, T)) = \mathit{prob}(\mathcal{SC}_{\leq\theta}(P_2, T))$$
■

We now prove the following properties of Markovian testing equivalence:

(1) $\sim_{\mathrm{MT}}$ is a strict refinement of nondeterministic testing equivalence [12].
(2) $\sim_{\mathrm{MT}}$ is a strict refinement of probabilistic testing equivalence [9,10].
(3) $\sim_{\mathrm{MT}}$ is strictly coarser than $\sim_{\mathrm{MB}}$.

Definition 4.37 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $T \in \mathcal{T}$. We say that:

- P may pass T, written $P\, may\, T$, iff at least one maximal test-driven computation is successful:
$$\mathcal{SC}(P, T) \neq \emptyset$$
- P must pass T, written $P\, must\, T$, iff all maximal test-driven computations are successful:
$$\mathcal{SC}(P, T) = \mathcal{I}_{\mathrm{f}}(P \,\|_{Name}\, T)$$ ■

Definition 4.38 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that:

- P_1 is may-testing equivalent to P_2, written $P_1 \sim_{\mathrm{T,may}} P_2$, iff for all tests $T \in \mathcal{T}$:
$$P_1\, may\, T \iff P_2\, may\, T$$
- P_1 is must-testing equivalent to P_2, written $P_1 \sim_{\mathrm{T,must}} P_2$, iff for all tests $T \in \mathcal{T}$:
$$P_1\, must\, T \iff P_2\, must\, T$$
- P_1 is testing equivalent to P_2, written $P_1 \sim_{\mathrm{T}} P_2$, iff:
$$P_1 \sim_{\mathrm{T,may}} P_2 \;\wedge\; P_1 \sim_{\mathrm{T,must}} P_2$$ ■

Lemma 4.39 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $T \in \mathcal{T}$. Then:

(1) $P\, may\, T$ iff there exist $p \in \mathbb{R}_{]0,1]}$ and $\theta \in \mathbb{R}^*_{>0}$ such that $P\, pass_{p,\theta}\, T$.
(2) $P\, must\, T$ iff there exists $\theta \in \mathbb{R}^*_{>0}$ such that $P\, pass_{1,\theta}\, T$.

Proof (1) $P\, may\, T$ means that there is at least one computation in $\mathcal{SC}(P \,\|_{Name}\, T)$, which will have execution probability $p \in \mathbb{R}_{]0,1]}$ and average duration $\theta \in \mathbb{R}^*_{>0}$. This amounts to say that $P\, pass_{p,\theta}\, T$.
(2) $P\, must\, T$ means that all the computations in $\mathcal{I}_{\mathrm{f}}(P \,\|_{Name}\, T)$ are successful. Since $\mathcal{I}_{\mathrm{f}}(P \,\|_{Name}\, T)$ is finite, $P\, pass_{1,\theta}\, T$ for $\theta \in \mathbb{R}^*_{>0}$ built from the element-by-element maximum of the average durations of such computations. ■

Proposition 4.40 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:
$$P_1 \sim_{\mathrm{MT}} P_2 \implies P_1 \sim_{\mathrm{T}} P_2$$

Proof We have to prove that from $P_1 \sim_{\mathrm{MT}} P_2$ it follows that $P_1 \sim_{\mathrm{T,may}} P_2 \wedge P_1 \sim_{\mathrm{T,must}} P_2$. Let $k \in \{1, 2\}$ and $h \in \{1, 2\} - \{k\}$:

- If $P_k \, may \, T$ for an arbitrary $T \in \mathcal{T}$, then by Lemma 4.39(1) there exist $p \in \mathbb{R}_{]0,1]}$ and $\theta \in \mathbb{R}^*_{>0}$ such that $P_k \, pass_{p,\theta} \, T$. Thus $P_h \, pass_{p,\theta} \, T$ by $P_1 \sim_{\rm MT} P_2$ and Prop. 4.12, hence $P_h \, may \, T$ by Lemma 4.39(1).
- If $P_k \, must \, T$ for an arbitrary $T \in \mathcal{T}$, then by Lemma 4.39(2) there exists $\theta \in \mathbb{R}^*_{>0}$ such that $P_k \, pass_{1,\theta} \, T$. Thus $P_h \, pass_{1,\theta} \, T$ by $P_1 \sim_{\rm MT} P_2$ and Prop. 4.12, hence $P_h \, must \, T$ by Lemma 4.39(2). ■

Example 4.41 The converse of Prop. 4.40 does not hold, i.e. nondeterministic testing equivalence is strictly coarser than Markovian testing equivalence. If we consider:

$$P_1 \equiv <a, \lambda>.\underline{0} + <b, \mu>.\underline{0}$$

$$P_2 \equiv <a, \mu>.\underline{0} + <b, \lambda>.\underline{0}$$

with $a \neq b$ and $\lambda \neq \mu$, then $P_1 \sim_{\rm T} P_2$ because both process terms can only perform an a-action and a b-action, but $P_1 \not\sim_{\rm MT} P_2$. For instance, the two process terms are distinguished by test:

$$T \equiv <a, *_1>.{\rm s} + <b, *_1>.{\rm f}$$

In fact, although $P_1 \, \|_{Name} \, T$ and $P_2 \, \|_{Name} \, T$ have the same average sojourn time $t = 1/(\lambda + \mu)$, the probability of passing T within t time units on average is $\lambda/(\lambda + \mu)$ in the case of P_1, while it is $\mu/(\lambda + \mu)$ in the case of P_2. ■

Definition 4.42 Let $P \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, and $p \in \mathbb{R}_{]0,1]}$. We say that P passes T with probability at least p, written $P \, pass_p \, T$, iff:

$$prob(\mathcal{SC}(P, T)) \geq p$$

■

Definition 4.43 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. We say that P_1 is probabilistic testing equivalent to P_2, written $P_1 \sim_{\rm PT} P_2$, iff for all tests $T \in \mathcal{T}$ and probabilities $p \in \mathbb{R}_{]0,1]}$:

$$P_1 \, pass_p \, T \iff P_2 \, pass_p \, T$$

■

Lemma 4.44 Let $P \in \mathcal{P}_{\rm S}$, $T \in \mathcal{T}$, and $p \in \mathbb{R}_{]0,1]}$. Then $P \, pass_p \, T$ iff there exists $\theta \in \mathbb{R}^*_{>0}$ such that $P \, pass_{p,\theta} \, T$.

Proof $P \, pass_p \, T$ means that the sum of the execution probabilities of all the computations in $\mathcal{SC}(P, T)$ is at least p. Since $\mathcal{SC}(P, T)$ is finite, $P \, pass_{p,\theta} \, T$ for $\theta \in \mathbb{R}^*_{>0}$ built from the element-by-element maximum of the average durations of such computations. ■

Proposition 4.45 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Then:

$$P_1 \sim_{\rm MT} P_2 \implies P_1 \sim_{\rm PT} P_2$$

Proof Let $k \in \{1, 2\}$ and $h \in \{1, 2\} - \{k\}$. If $P_k \, pass_p \, T$ for an arbitrary $T \in \mathcal{T}$ and an arbitrary $p \in \mathbb{R}_{]0,1]}$, then by Lemma 4.44 there exists $\theta \in \mathbb{R}^*_{>0}$

such that $P_k \; pass_{p,\theta} \; T$. Thus $P_h \; pass_{p,\theta} \; T$ by $P_1 \sim_{\rm MT} P_2$ and Prop. 4.12, hence $P_h \; pass_p \; T$ by Lemma 4.44. ■

Example 4.46 The converse of Prop. 4.45 does not hold, i.e. probabilistic testing equivalence is strictly coarser than Markovian testing equivalence.
If we consider:

$$P_3 \equiv <a, \lambda>.\underline{0} + <b, \mu>.\underline{0}$$

$$P_4 \equiv <a, 2 \cdot \lambda>.\underline{0} + <b, 2 \cdot \mu>.\underline{0}$$

with $a \neq b$, then $P_3 \sim_{\rm PT} P_4$ because both process terms can only perform an a-action and a b-action with the same probabilities – $\lambda/(\lambda+\mu)$ and $\mu/(\lambda+\mu)$, respectively – but $P_3 \not\sim_{\rm MT} P_4$. For instance, the two process terms are distinguished by test:

$$T \equiv <a, *_1>.{\rm s} + <b, *_1>.{\rm f}$$

In fact, the average sojourn time of $P_3 \,\|_{Name}\, T$ is $1/(\lambda+\mu)$ whereas the average sojourn time of $P_4 \,\|_{Name}\, T$ is $1/(2 \cdot \lambda + 2 \cdot \mu)$, hence the probability of passing T within $1/(2 \cdot \lambda + 2 \cdot \mu)$ time units on average is 0 in the case of P_3, while it is $\lambda/(\lambda + \mu)$ in the case of P_4. ■

Corollary 4.47 $\sim_{\rm MT} \subset \sim_{\rm PT} \subset \sim_{\rm T}$. ■

Proposition 4.48 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Then:

$$P_1 \sim_{\rm MB} P_2 \implies P_1 \sim_{\rm MT} P_2$$

Proof We proceed by induction on the syntactical structure of an arbitrary test T in order to prove that from $P_1 \sim_{\rm MB} P_2$ it follows $prob(\mathcal{SC}_{\leq\theta}(P_1, T)) = prob(\mathcal{SC}_{\leq\theta}(P_2, T))$ for all $\theta \in \mathbb{R}^*_{>0}$:

- If $T \equiv {\rm f}$ then for all $\theta \in \mathbb{R}^*_{>0}$:
$$prob(\mathcal{SC}_{\leq\theta}(P_1, T)) = 0 = prob(\mathcal{SC}_{\leq\theta}(P_2, T))$$
- If $T \equiv {\rm s}$ then for all $\theta \in \mathbb{R}^*_{>0}$:
$$prob(\mathcal{SC}_{\leq\theta}(P_1, T)) = 1 = prob(\mathcal{SC}_{\leq\theta}(P_2, T))$$
- Let $T \equiv \sum_{i \in I} <a_i, *_{w_i}>.T_i$ with I finite and non-empty. From $P_1 \sim_{\rm MB} P_2$ it follows that $P_1 \,\|_{Name}\, T \sim_{\rm MB} P_2 \,\|_{Name}\, T$ by virtue of Thm. 3.3(3), hence for some $r \in \mathbb{R}_{\geq 0}$:
$$rate_{\rm t}(P_1 \,\|_{Name}\, T, 0) = r = rate_{\rm t}(P_2 \,\|_{Name}\, T, 0)$$
Let $\theta \in \mathbb{R}^*_{>0}$. If $r = 0$ or $length(\theta) = 0$ or $\theta[1] < \frac{1}{r}$, then:
$$prob(\mathcal{SC}_{\leq\theta}(P_1, T)) = 0 = prob(\mathcal{SC}_{\leq\theta}(P_2, T))$$
If instead $r > 0$ and $\theta = \frac{1}{\mu} \circ \theta'$ with $\frac{1}{\mu} \geq \frac{1}{r}$, then for each $k = 1, 2$:
$$prob(\mathcal{SC}_{\leq\theta}(P_k, T)) = \sum_{i \in I} \sum_{P_k \xrightarrow{a_i, \lambda} Q} \frac{\lambda \cdot w_i / weight(T, a_i)}{r} \cdot prob(\mathcal{SC}_{\leq\theta'}(Q, T_i))$$
By applying the induction hypothesis to all the process terms in the same

equivalence class $[Q]$ with respect to $\sim_{\rm MB}$ when tested against T_i for any $i \in I$, we derive:

$$prob(\mathcal{SC}_{\leq \theta}(P_k, T)) = \sum_{i \in I} \sum_{[Q] \in \mathcal{P}_{\rm S}/\sim_{\rm MB}} \frac{rate(P_k, a_i, 0, [Q]) \cdot w_i / weight(T, a_i)}{r} \cdot prob(\mathcal{SC}_{\leq \theta'}(Q, T_i))$$

From $P_1 \sim_{\rm MB} P_2$ it follows that for all $i \in I$ and $[Q] \in \mathcal{P}_{\rm S}/\sim_{\rm MB}$:

$$rate(P_1, a_i, 0, [Q]) = rate(P_2, a_i, 0, [Q])$$

hence the result. ∎

Example 4.49 The converse of Prop. 4.48 does not hold, i.e. Markovian testing equivalence is strictly coarser than Markovian bisimilarity.
If we consider:

$$P_5 \equiv <a, \lambda_1>.<b, \mu>.P' + <a, \lambda_2>.<b, \mu>.P''$$

$$P_6 \equiv <a, \lambda_1 + \lambda_2>.(<b, \tfrac{\lambda_1}{\lambda_1+\lambda_2} \cdot \mu>.P' + <b, \tfrac{\lambda_2}{\lambda_1+\lambda_2} \cdot \mu>.P'')$$

then we have $P_5 \sim_{\rm MT} P_6$ but $P_5 \not\sim_{\rm MB} P_6$ if $P' \not\sim_{\rm MB} P''$. In fact, no test starting with a passive a-action possibly followed by a passive b-action can distinguish between the two terms, because in both terms the average time to perform an a-action followed by a b-action is $1/(\lambda_1 + \lambda_2) \circ 1/\mu$ and the probability of reaching P' (resp. P'') is $\lambda_1/(\lambda_1 + \lambda_2)$ (resp. $\lambda_2/(\lambda_1 + \lambda_2)$). By contrast, there is no way to relate $<b, \mu>.P'$ and $<b, \mu>.P''$ with $<b, \lambda_1/(\lambda_1 + \lambda_2) \cdot \mu>.P' + <b, \lambda_2/(\lambda_1 + \lambda_2) \cdot \mu>.P''$ through $\sim_{\rm MB}$ whenever $P' \not\sim_{\rm MB} P''$. ∎

4.5 Congruence Property

We now show that Markovian testing equivalence turns out to be a congruence with respect to all the operators of SMPC.

Lemma 4.50 Let $P \in \mathcal{P}_{\rm S}$ and $T \in \mathcal{T}$. Then for all $a \in Name$, $\lambda, w \in \mathbb{R}_{>0}$, and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq \theta}(P, T)) = prob(\mathcal{SC}_{\leq \frac{1}{\lambda} \circ \theta}(<a, \lambda>.P, <a, *_w>.T))$$

Proof It follows from the fact that a finite-length computation c belongs to $\mathcal{SC}(P, T)$ iff $<a, \lambda>.P \,\|_{Name} <a, *_w>.T \xrightarrow{a, \lambda} c$ belongs to $\mathcal{SC}(<a, \lambda>.P, <a, *_w>.T)$, with the average time spent before executing the additional initial transition equal to $1/\lambda$. ∎

Lemma 4.51 Let $P \in \mathcal{P}_{\rm S}$, $T \equiv \sum_{i \in I} <a_i, *_{w_i}>.T_i \in \mathcal{T}$, and $a \in Name$ such that $weight(T, a) > 0$. Then for all $\lambda \in \mathbb{R}_{>0}$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq \theta}(<a, \lambda>.P, T)) = \sum_{i \in I_a} \frac{w_i}{weight(T, a)} \cdot prob(\mathcal{SC}_{\leq \theta}(<a, \lambda>.P, <a_i, *_{w_i}>.T_i))$$

where $I_a = \{i \in I \mid a_i = a\}$.

Proof The only summands of T that can interact with $<a, \lambda>.P$ are those initially enabling a-actions, hence the restriction to I_a. The result stems from the fact that $<a, \lambda>.P \parallel_{Name} T \xrightarrow{a, \lambda \cdot w_i / weight(T,a)} c$ belongs to $\mathcal{SC}(<a, \lambda>.P, T)$ iff $<a, \lambda>.P \parallel_{Name} <a_i, *_{w_i}>.T_i \xrightarrow{a, \lambda} c$ belongs to $\mathcal{SC}(<a, \lambda>.P, <a_i, *_{w_i}>.T_i)$ for some $i \in I_a$, with the probability of executing the initial transition equal to $w_i / weight(T, a)$ in the first case and the average time spent before executing the initial transition equal to $1/\lambda$ in both cases. ■

Lemma 4.52 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$ and $T \in \mathcal{T} - \{{\rm f}, {\rm s}\}$. Then for all $\theta \in \mathbb{R}^{+}_{>0}$:

$$prob(\mathcal{SC}_{\le \theta}(P_1 + P_2, T)) = \begin{cases} p_1 \cdot prob(\mathcal{SC}_{\le \theta_1}(P_1, T)) + p_2 \cdot prob(\mathcal{SC}_{\le \theta_2}(P_2, T)) \\ \qquad \text{if } r_1 > 0 \wedge r_2 > 0 \\ prob(\mathcal{SC}_{\le \theta}(P_1, T)) \\ \qquad \text{if } r_1 > 0 \wedge r_2 = 0 \\ prob(\mathcal{SC}_{\le \theta}(P_2, T)) \\ \qquad \text{if } r_1 = 0 \wedge r_2 > 0 \\ 0 \\ \qquad \text{if } r_1 = 0 \wedge r_2 = 0 \end{cases}$$

where:

$$r_1 = rate_{\rm t}(P_1 \parallel_{Name} T, 0) \qquad r_2 = rate_{\rm t}(P_2 \parallel_{Name} T, 0)$$

$$p_1 = \tfrac{r_1}{r_1 + r_2} \qquad p_2 = \tfrac{r_2}{r_1 + r_2}$$

$$\theta_1[i] = \begin{cases} \theta[i] + (\frac{1}{r_1} - \frac{1}{r_1 + r_2}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases} \qquad \theta_2[i] = \begin{cases} \theta[i] + (\frac{1}{r_2} - \frac{1}{r_1 + r_2}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases}$$

Proof Observed that the computations in $\mathcal{SC}(P_1 + P_2, T)$ differ from those in $\mathcal{SC}(P_1, T) \cup \mathcal{SC}(P_2, T)$ only for their initial state, we concentrate on the first case, as the other three are trivial. Therefore, we assume that both $P_1 \parallel_{Name} T$ and $P_2 \parallel_{Name} T$ can execute at least one action. There are two aspects to be taken into account.
From the probabilistic viewpoint, the computations in $\mathcal{SC}(P_1 + P_2, T)$ differ from those in $\mathcal{SC}(P_1, T) \cup \mathcal{SC}(P_2, T)$ only for the probability of executing their first transition. More precisely, if $c \in \mathcal{SC}(P_k, T)$ for some $k \in \{1, 2\}$, then the probability of c computed in the interaction system of $P_1 + P_2$ and T is the probability of c computed in the interaction system of P_k and T multiplied by a factor representing the fact that the first transition of c stems from an action of P_k. Such a factor is exactly p_k.
From the timing viewpoint, the computations of $\mathcal{SC}(P_1 + P_2, T)$ differ from those of $\mathcal{SC}(P_1, T) \cup \mathcal{SC}(P_2, T)$ only for the average time spent in the ini-

tial state before the execution of their first transition. More precisely, if $c \in \mathcal{SC}(P_k, T)$ for some $k \in \{1, 2\}$, then the average time spent in the initial state is $1/r_k$ in the interaction system of P_k and T, whereas it is $1/(r_1 + r_2)$ in the interaction system of $P_1 + P_2$ and T. This means that on average P_k takes more time to reach success than $P_1 + P_2$, with the extra average time equal to $1/r_k - 1/(r_1 + r_2)$. This justifies the definition of $\theta_k[i]$ for $i = 1$. ■

Theorem 4.53 Let $P_1, P_2 \in \mathcal{P}_S$. Whenever $P_1 \sim_{MT} P_2$, then:

(1) $<a, \lambda>.P_1 \sim_{MT} <a, \lambda>.P_2$ for all $<a, \lambda> \in Act_S$.
(2) $P_1 + P \sim_{MT} P_2 + P$ and $P + P_1 \sim_{MT} P + P_2$ for all $P \in \mathcal{P}_S$.

Proof (1) In order to avoid trivial cases, consider $T \equiv \sum_{i \in I} <a_i, *_{w_i}>.T_i \in \mathcal{T}$ with $I_a = \{i \in I \mid a_i = a\} \neq \emptyset$. From $P_1 \sim_{MT} P_2$ it follows that for all $i \in I_a$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq \theta}(P_1, T_i)) = prob(\mathcal{SC}_{\leq \theta}(P_2, T_i))$$

By virtue of Lemma 4.50, for each $k = 1, 2$ we have that for all $i \in I_a$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{SC}_{\leq \theta}(P_k, T_i)) = prob(\mathcal{SC}_{\leq \frac{1}{\lambda} \circ \theta}(<a, \lambda>.P_k, <a_i, *_{w_i}>.T_i))$$

Then for all $i \in I_a$ and $\theta' = \frac{1}{\mu} \circ \theta \in \mathbb{R}^+_{>0}$ such that $\frac{1}{\mu} \geq \frac{1}{\lambda}$ we have:

$$prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_1, <a_i, *_{w_i}>.T_i)) = prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_2, <a_i, *_{w_i}>.T_i))$$

hence:

$$\sum_{i \in I_a} \frac{w_i}{weight(T, a)} \cdot prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_1, <a_i, *_{w_i}>.T_i)) =$$
$$\sum_{i \in I_a} \frac{w_i}{weight(T, a)} \cdot prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_2, <a_i, *_{w_i}>.T_i))$$

from which, by virtue of Lemma 4.51, we derive that:

$$prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_1, T)) = prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_2, T))$$

Since $T \not\equiv \mathrm{s}$, for $\theta' = \varepsilon$ and for all $\theta' \in \mathbb{R}^+_{>0}$ such that $\theta'[1] < \frac{1}{\lambda}$ we have:

$$prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_1, T)) = 0 = prob(\mathcal{SC}_{\leq \theta'}(<a, \lambda>.P_2, T))$$

(2) In order to avoid trivial cases, consider $T \in \mathcal{T} - \{\mathrm{f}, \mathrm{s}\}$. By virtue of Lemma 4.52, for each $k = 1, 2$ we have that for all $\theta \in \mathbb{R}^+_{>0}$:

$$prob(\mathcal{SC}_{\leq \theta}(P_k + P, T)) = \begin{cases} p_k \cdot prob(\mathcal{SC}_{\leq \theta_k}(P_k, T)) + p'_k \cdot prob(\mathcal{SC}_{\leq \theta'_k}(P, T)) \\ \qquad \text{if } r_k > 0 \wedge r > 0 \\ prob(\mathcal{SC}_{\leq \theta}(P_k, T)) \\ \qquad \text{if } r_k > 0 \wedge r = 0 \\ prob(\mathcal{SC}_{\leq \theta}(P, T)) \\ \qquad \text{if } r_k = 0 \wedge r > 0 \\ 0 \\ \qquad \text{if } r_k = 0 \wedge r = 0 \end{cases}$$

where:

$$r_k = rate_t(P_k \,\|_{Name}\, T, 0) \qquad r = rate_t(P \,\|_{Name}\, T, 0)$$

$$p_k = \frac{r_k}{r_k+r} \qquad p'_k = \frac{r}{r_k+r}$$

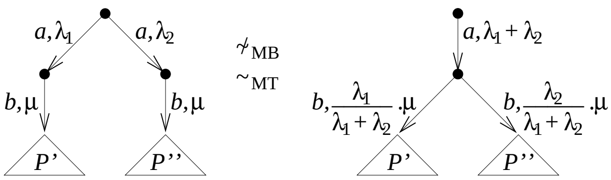
$$\theta_k[i] = \begin{cases} \theta[i] + (\frac{1}{r_k} - \frac{1}{r_k+r}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases} \qquad \theta'_k[i] = \begin{cases} \theta[i] + (\frac{1}{r} - \frac{1}{r_k+r}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases}$$

By virtue of Cor. 4.8, from $P_1 \sim_{\rm MT} P_2$ it follows that $r_1 = r_2$. Therefore $p_1 = p_2$, $p'_1 = p'_2$, $\theta_1 = \theta_2$, and $\theta'_1 = \theta'_2$. From $P_1 \sim_{\rm MT} P_2$ it also follows that $prob(\mathcal{SC}_{\leq \theta_1}(P_1, T)) = prob(\mathcal{SC}_{\leq \theta_2}(P_2, T))$ and $prob(\mathcal{SC}_{\leq \theta}(P_1, T)) = prob(\mathcal{SC}_{\leq \theta}(P_2, T))$, hence $prob(\mathcal{SC}_{\leq \theta}(P_1 + P, T)) = prob(\mathcal{SC}_{\leq \theta}(P_2 + P, T))$. Since $T \not\equiv \mathrm{s}$, for $\theta = \varepsilon$ we have:

$$prob(\mathcal{SC}_{\leq \theta}(P_1 + P, T)) = 0 = prob(\mathcal{SC}_{\leq \theta}(P_2 + P, T)) \qquad \blacksquare$$

4.6 Sound and Complete Axiomatization

As shown by Prop. 4.48 and Ex. 4.49, $\sim_{\rm MB}$ is strictly contained in $\sim_{\rm MT}$, hence the axioms $\mathcal{A}_1^{\rm MB}$-$\mathcal{A}_4^{\rm MB}$ of Table 1 are still valid for $\sim_{\rm MT}$ over $\mathcal{P}_{\rm S,nr}$, but not complete. In fact, the two process terms considered in Ex. 4.49, which are depicted below:



show that $\sim_{\rm MB}$ is highly sensitive to branching points. By contrast, $\sim_{\rm MT}$ allows choices to be deferred as long as they are related to branches starting with actions having the same name that are immediately followed by actions having the same names and the same total rates in all the branches.

Here we prove that the two terms above constitute the simplest instance of an axiom schema subsuming $\mathcal{A}_4^{\rm MB}$ that we have to introduce in order to obtain a sound and complete axiomatization of $\sim_{\rm MT}$ over $\mathcal{P}_{\rm S,nr}$. Such an axiomatization is given by the set $\mathcal{A}^{\rm MT}$ of axioms shown in Table 2, where I and J_i are finite index sets with $|I| \geq 2$ (if $J_i = \emptyset$, the related summations are taken to be $\underline{0}$).

Theorem 4.54 The deduction system $DED(\mathcal{A}^{\rm MT})$ is sound for $\sim_{\rm MT}$ over $\mathcal{P}_{\rm S,nr}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\rm S,nr}$:

$$\mathcal{A}^{\rm MT} \vdash P_1 = P_2 \implies P_1 \sim_{\rm MT} P_2$$

Proof Since $\sim_{\rm MT}$ is an equivalence relation and a congruence with respect to action prefix and alternative composition by virtue of Thm. 4.53, in the proof

$(\mathcal{A}_1^{\mathrm{MT}})$	$P_1 + P_2 = P_2 + P_1$
$(\mathcal{A}_2^{\mathrm{MT}})$	$(P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$
$(\mathcal{A}_3^{\mathrm{MT}})$	$P + \underline{0} = P$
$(\mathcal{A}_4^{\mathrm{MT}})$	$\sum\limits_{i \in I} <a, \lambda_i>. \sum\limits_{j \in J_i} <b_{i,j}, \mu_{i,j}>.P_{i,j} = \quad <a, \sum_{k \in I} \lambda_k>. \sum\limits_{i \in I} \sum\limits_{j \in J_i} <b_{i,j}, \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \mu_{i,j}>.P_{i,j}$ if for all $i_1, i_2 \in I$: $\{b_{i_1,j} \mid j \in J_{i_1}\} = \{b_{i_2,j} \mid j \in J_{i_2}\} \equiv \{b_1, b_2, \ldots, b_n\}$ and for all $h = 1, \ldots, n$: $\sum\limits_{j \in J_{i_1}} \{\!\mid \mu_{i_1,j} \mid b_{i_1,j} = b_h \mid\!\} = \sum\limits_{j \in J_{i_2}} \{\!\mid \mu_{i_2,j} \mid b_{i_2,j} = b_h \mid\!\} \equiv \mu_h$

Table 2
Axiomatization of $\sim_{\mathrm{MT}}$ over $\mathcal{P}_{\mathrm{S,nr}}$

of $\mathcal{A}^{\mathrm{MT}} \vdash P_1 = P_2$ it is correct to use reflexivity, symmetry, transitivity, and substitutivity with respect to action prefix and alternative composition.
As far as the set $\mathcal{A}^{\mathrm{MT}}$ of specific axioms is concerned, from Prop. 4.48 it follows that $\sim_{\mathrm{MT}}$ inherits all the axioms of $\sim_{\mathrm{MB}}$, hence $\mathcal{A}_1^{\mathrm{MT}}$-$\mathcal{A}_3^{\mathrm{MT}}$ are certainly correct for $\sim_{\mathrm{MT}}$. For $\mathcal{A}_4^{\mathrm{MT}}$ it suffices to observe what follows:

- Both terms can initially execute only a-actions.
- The average time to execute them is $1/\sum_{k \in I} \lambda_k$ in both terms.
- If $J_i = \emptyset$ for all $i \in I$, then the a-derivative term is $\underline{0}$ with probability 1 both on the left and on the right, so no test can distinguish between the two original terms. If instead $J_i \neq \emptyset$ for all $i \in I$, then the a-derivative term is $\sum_{j \in J_i} <b_{i,j}, \mu_{i,j}>.P_{i,j}$ with probability $\lambda_i/\sum_{k \in I} \lambda_k$ on the left, while it is $\sum_{i \in I} \sum_{j \in J_i} <b_{i,j}, \lambda_i/\sum_{k \in I} \lambda_k \cdot \mu_{i,j}>.P_{i,j}$ with probability 1 on the right. Not even at this point can a test make a distinction because:
  - · All the a-derivative terms can initially execute only b_h-actions for $h \in \{1, \ldots, n\}$.
  - · Each of the b_h-actions has the same total rate μ_h in all the a-derivative terms.
  - · For all tests initially enabling a followed by one or more alternatives among which $b_{i,j}$, the a-$b_{i,j}$-derivative term $P_{i,j}$ is reached with the same probability $(\lambda_i/\sum_{k \in I} \lambda_k) \cdot (\mu_{i,j}/\sum_{h=1}^{n} \{\!\mid \mu_h \mid b_h \text{ test-enabled} \mid\!\})$ both on the left and on the right. ■

Definition 4.55 Let $P \in \mathcal{P}_{\mathrm{S,nr}}$. We say that P is in testing-minimal sum normal form (tmsnf) iff $P \equiv \underline{0}$ or $P \equiv \sum_{i \in I} <a_i, \lambda_i>.P_i$ with I finite and non-empty, P initially minimal with respect to $\mathcal{A}_4^{\mathrm{MT}}$, and P_i in tmsnf for all $i \in I$. ■

In the previous inductive definition, by initial minimality of P with respect to $\mathcal{A}_4^{\rm MT}$ we mean that no subset of summands of P matches the left-hand side term of $\mathcal{A}_4^{\rm MT}$. From the definition it follows that the initial minimality holds with respect to $\mathcal{A}_3^{\rm MT}$ as well. We denote by $\mathcal{P}_{\rm S,nr,tmsnf}$ the set of the non-recursive process terms of $\mathcal{P}_{\rm S}$ that are in tmsnf.

Lemma 4.56 For all $P \in \mathcal{P}_{\rm S,nr}$ there exists $Q \in \mathcal{P}_{\rm S,nr,tmsnf}$ such that $\mathcal{A}^{\rm MT} \vdash P = Q$.

Proof We proceed by induction on the syntactical structure of the non-recursive process term P:

- If $P \equiv \underline{0}$, the result follows by taking $Q \equiv \underline{0}$ (which is in tmsnf) and using reflexivity.
- If $P \equiv {<}a, \lambda{>}.P'$, then by the induction hypothesis there exists Q' in tmsnf such that $\mathcal{A}^{\rm MT} \vdash P' = Q'$. From substitutivity with respect to action prefix we obtain that $\mathcal{A}^{\rm MT} \vdash {<}a, \lambda{>}.P' = {<}a, \lambda{>}.Q'$, from which the result follows as ${<}a, \lambda{>}.Q'$ is in tmsnf.
- If $P \equiv P_1 + P_2$, then by the induction hypothesis there exist Q_1 and Q_2 in tmsnf such that $\mathcal{A}^{\rm MT} \vdash P_1 = Q_1$ and $\mathcal{A}^{\rm MT} \vdash P_2 = Q_2$. From substitutivity with respect to alternative composition we obtain that $\mathcal{A}^{\rm MT} \vdash P_1 + P_2 = Q_1 + Q_2$. If $Q_1 + Q_2$ is in tmsnf, we are done. If instead $Q_1 + Q_2$ is not in tmsnf – because it is not initially minimal with respect to $\mathcal{A}_3^{\rm MT}$ or $\mathcal{A}_4^{\rm MT}$ – the result follows after as many applications of $\mathcal{A}_3^{\rm MT}$ and $\mathcal{A}_4^{\rm MT}$ to Q_1+Q_2 as possible (possibly preceded by applications of $\mathcal{A}_1^{\rm MT}$ and $\mathcal{A}_2^{\rm MT}$) by virtue of substitutivity with respect to alternative composition as well as transitivity.

■

Theorem 4.57 The deduction system $DED(\mathcal{A}^{\rm MT})$ is complete for $\sim_{\rm MT}$ over $\mathcal{P}_{\rm S,nr}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\rm S,nr}$:

$$P_1 \sim_{\rm MT} P_2 \implies \mathcal{A}^{\rm MT} \vdash P_1 = P_2$$

Proof There are two cases. If P_1 and P_2 are both in tmsnf, we proceed by induction on the syntactical structure of P_1:

- If $P_1 \equiv \underline{0}$, from $P_1 \sim_{\rm MT} P_2$ and P_2 in tmsnf it follows that $P_2 \equiv \underline{0}$, hence the result by reflexivity.
- If $P_1 \equiv \sum_{i \in I_1} {<}a_i, \lambda_i{>}.P_{1,i}$ with I_1 finite and non-empty, from $P_1 \sim_{\rm MT} P_2$ and P_2 in tmsnf it follows that $P_2 \equiv \sum_{j \in I_2} {<}b_j, \mu_j{>}.P_{2,j}$ with I_2 finite and non-empty. By virtue of Cor. 4.8, from $P_1 \sim_{\rm MT} P_2$ we derive that:

  $$\{a_i \mid i \in I_1\} = \{b_j \mid j \in I_2\} \equiv \{c_1, c_2, \dots, c_n\}$$

  with:

  $$rate(P_1, c_k, 0, \mathcal{P}_{\rm C}) = rate(P_2, c_k, 0, \mathcal{P}_{\rm C})$$

  for each $k = 1, \dots, n$. We can then concentrate on a generic c_k and on the two related sets of summands:

$$S_{k,1} = \{<\!a_i, \lambda_i\!>.P_{1,i} \mid i \in I_1 \wedge a_i = c_k\}$$

$$S_{k,2} = \{<\!b_j, \mu_j\!>.P_{2,j} \mid j \in I_2 \wedge b_j = c_k\}$$

which satisfy the following two properties:

(1) $\sum_{P \in S_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}}) = \sum_{P \in S_{k,2}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}})$.

(2) The derivative terms $P_{1,i}$ (resp. $P_{2,j}$) occurring in $S_{k,1}$ (resp. $S_{k,2}$) are all inequivalent with respect to $\sim_{\mathrm{MT}}$ due to the initial minimality of P_1 (resp. P_2) with respect to $\mathcal{A}_4^{\mathrm{MT}}$. In fact, due to such an initial minimality, taken two derivative terms in the same summand set, it must be the case that their sets of initial action names are different or the total exit rate with respect to one of these initial action names is different in the two derivative terms, thus violating the necessary condition for $\sim_{\mathrm{MT}}$ stated by Cor. 4.8.

We now prove by proceeding by induction on $|S_{k,1}|$ that for each summand $<\!a_i, \lambda_i\!>.P_{1,i} \in S_{k,1}$ there exists exactly one summand $<\!b_j, \mu_j\!>.P_{2,j} \in S_{k,2}$ such that $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\mathrm{MT}} P_{2,j}$:

- If $|S_{k,1}| = 1$, then $S_{k,1}$ contains a single summand, say $<\!a_i, \lambda_i\!>.P_{1,i}$. Then $S_{k,2}$ must contain a single summand as well, say $<\!b_j, \mu_j\!>.P_{2,j}$, with $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\mathrm{MT}} P_{2,j}$ (as $P_1 \sim_{\mathrm{MT}} P_2$, hence P_1 and P_2 cannot be distinguished by tests starting with a c_k-action). The reason why $S_{k,2}$ cannot contain several summands – each starting with a c_k-action – is that this would contradict $P_1 \sim_{\mathrm{MT}} P_2$ (in the case in which the inequivalent derivatives of the summands violated the necessary condition for $\sim_{\mathrm{MT}}$ stated by Cor. 4.8) or the initial minimality of P_2 with respect to $\mathcal{A}_4^{\mathrm{MT}}$ (in the case in which the inequivalent derivatives satisfied that necessary condition).
- Suppose that $|S_{k,1}| = m > 1$. Let $S^d_{k,1}$ be the set of the summands of $S_{k,1}$ whose derivative terms have – among all the derivative terms occurring in $S_{k,1}$ – the maximum total exit rate δ with respect to an action name d. By virtue of property (2), d can be chosen in such a way that $S^d_{k,1} \neq S_{k,1}$. Then the derivative term of each summand of $S^d_{k,1}$ passes with probability 1 the test $<\!d, *_1\!>.\mathrm{s}$ within the minimum average time $1/\delta$, hence P_1 passes with probability $\sum_{P \in S^d_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}}) / \sum_{P \in S_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}})$ the test $<\!c_k, *_1\!>.<\!d, *_1\!>.\mathrm{s}$ within the minimum average time sequence $1/\sum_{P \in S_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}}) \circ 1/\delta$. Since $P_1 \sim_{\mathrm{MT}} P_2$, also P_2 must pass the same test in the same way as P_1, hence there must exist a subset $S^d_{k,2}$ of $S_{k,2}$ whose derivative terms all have the maximum total exit rate δ with respect to d, with $S^d_{k,2} \neq S_{k,2}$ and $\sum_{P \in S^d_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}}) = \sum_{P \in S^d_{k,2}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}})$.

  Since $S^d_{k,1}$ and $S^d_{k,2}$ satisfy properties (1) and (2) with $|S^d_{k,1}| < m$, by the induction hypothesis it follows that for each summand $<\!a_i, \lambda_i\!>.P_{1,i} \in S^d_{k,1}$ there exists exactly one summand $<\!b_j, \mu_j\!>.P_{2,j} \in S^d_{k,2}$ such that $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\mathrm{MT}} P_{2,j}$.

  Likewise, since $S'_{k,1} = S_{k,1} - S^d_{k,1}$ and $S'_{k,2} = S_{k,2} - S^d_{k,2}$ satisfy properties (1) and (2) with $|S'_{k,1}| < m$, by the induction hypothesis it follows that for each summand $<a_i, \lambda_i>.P_{1,i} \in S'_{k,1}$ there exists exactly one summand $<b_j, \mu_j>.P_{2,j} \in S'_{k,2}$ such that $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\rm MT} P_{2,j}$. Thus the result follows for the whole $S_{k,1}$ and $S_{k,2}$.

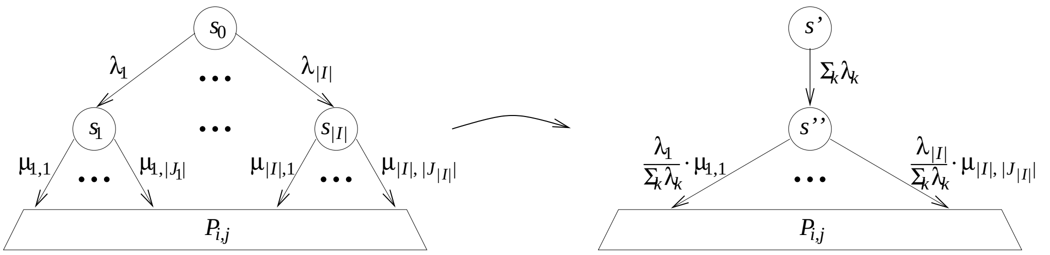
By proceeding in a similar way we can prove that for each summand $<b_j, \mu_j>.P_{2,j} \in S_{k,2}$ there exists exactly one summand $<a_i, \lambda_i>.P_{1,i} \in S_{k,1}$ such that $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\rm MT} P_{2,j}$.

As a consequence, a bijective correspondence can be established between $S_{k,1}$ and $S_{k,2}$. Let us now take a pair of corresponding summands $<a_i, \lambda_i>.P_{1,i}$ and $<b_j, \mu_j>.P_{2,j}$, so that $a_i = b_j \wedge \lambda_i = \mu_j \wedge P_{1,i} \sim_{\rm MT} P_{2,j}$. Since $P_{1,i}$ and $P_{2,j}$ are in tmsnf, by the induction hypothesis it follows that $\mathcal{A}^{\rm MT} \vdash P_{1,i} = P_{2,j}$, hence $\mathcal{A}^{\rm MT} \vdash <a_i, \lambda_i>.P_{1,i} = <b_j, \mu_j>.P_{2,j}$ by substitutivity with respect to action prefix, hence $\mathcal{A}^{\rm MT} \vdash \sum_{i \in I_1} <a_i, \lambda_i>.P_{1,i} = \sum_{j \in I_2} <b_j, \mu_j>.P_{2,j}$ by substitutivity with respect to alternative composition and the bijectivity of the correspondence.

If instead it is not the case that P_1 and P_2 are both in tmsnf, we exploit Lemma 4.56 in order to derive Q_1 and Q_2 in tmsnf such that $\mathcal{A}^{\rm MT} \vdash P_1 = Q_1$ and $\mathcal{A}^{\rm MT} \vdash P_2 = Q_2$. By virtue of Thm. 4.54 it follows that $P_1 \sim_{\rm MT} Q_1$ and $P_2 \sim_{\rm MT} Q_2$, hence $Q_1 \sim_{\rm MT} Q_2$ since $P_1 \sim_{\rm MT} P_2$ by the initial hypothesis and $\sim_{\rm MT}$ is a transitive relation. Since Q_1 and Q_2 are both in tmsnf, from what demonstrated in the previous part of this proof we obtain that $\mathcal{A}^{\rm MT} \vdash Q_1 = Q_2$, hence $\mathcal{A}^{\rm MT} \vdash P_1 = P_2$ by transitivity. ■

4.7 Exact Aggregation Property

The axiomatization of $\sim_{\rm MT}$ over $\mathcal{P}_{\rm S,nr}$ differs from the one of $\sim_{\rm MB}$ over $\mathcal{P}_{\rm S,nr}$ only for the last axiom, thus we can concentrate on $\mathcal{A}_4^{\rm MT}$ to study the aggregation induced by $\sim_{\rm MT}$ at the CTMC level. If we view $\mathcal{A}_4^{\rm MT}$ as the following rewriting rule:



where for all $i_1, i_2 \in I$:

$$\sum_{j \in J_{i_1}} \mu_{i_1,j} = \sum_{j \in J_{i_2}} \mu_{i_2,j} \equiv \mu$$

it turns out that $\mathcal{A}_4^{\rm MT}$ aggregates $|I| \geq 2$ states into a single one and, as a consequence, merges the $|I| \geq 2$ transitions entering the states $s_1, s_2, \ldots, s_{|I|}$ into a single transition entering the new state s''.

Now the question arises as to whether this kind of aggregation is exact, i.e. whether the transient/stationary probability of being in the macrostate s'' of the aggregated CTMC on the right is the sum of the transient/stationary probabilities of being in one of the constituent microstates $s_1, s_2, \dots, s_{|I|}$ of the original CTMC on the left. A positive answer would entail the meaningfulness of $\sim_{\rm MT}$ for performance evaluation purposes, i.e. the preservation of the value of the performance measures across process terms that are Markovian testing equivalent (the remark done at the end of Sect. 3.4 applies also here).

Theorem 4.58 The CTMC-level aggregation induced by $\sim_{\rm MT}$ is exact.

Proof With respect to the figure above, we denote by $\boldsymbol{\pi}_{\rm l}^{(t)}$ and $\boldsymbol{\pi}_{\rm l}$ (resp. $\boldsymbol{\pi}_{\rm r}^{(t)}$ and $\boldsymbol{\pi}_{\rm r}$) the vectors containing for each state of the original CTMC on the left (resp. the aggregated CTMC on the right) the transient probability at time $t \in \mathbb{R}_{\geq 0}$ and the stationary probability of being in that state. Assumed that $\boldsymbol{\pi}_{\rm l}^{(0)}$ is given and that $\boldsymbol{\pi}_{\rm r}^{(0)}$ is obtained from it by letting:

$$\begin{aligned}
\boldsymbol{\pi}_{\rm r}^{(0)}[s'] &= \boldsymbol{\pi}_{\rm l}^{(0)}[s_0] \\
\boldsymbol{\pi}_{\rm r}^{(0)}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\rm l}^{(0)}[s_i] \\
\boldsymbol{\pi}_{\rm r}^{(0)}[s] &= \boldsymbol{\pi}_{\rm l}^{(0)}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

what we have to prove is that for all $t \in \mathbb{R}_{>0}$:

$$\begin{aligned}
\boldsymbol{\pi}_{\rm r}^{(t)}[s'] &= \boldsymbol{\pi}_{\rm l}^{(t)}[s_0] \\
\boldsymbol{\pi}_{\rm r}^{(t)}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\rm l}^{(t)}[s_i] \\
\boldsymbol{\pi}_{\rm r}^{(t)}[s] &= \boldsymbol{\pi}_{\rm l}^{(t)}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

and that:

$$\begin{aligned}
\boldsymbol{\pi}_{\rm r}[s'] &= \boldsymbol{\pi}_{\rm l}[s_0] \\
\boldsymbol{\pi}_{\rm r}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\rm l}[s_i] \\
\boldsymbol{\pi}_{\rm r}[s] &= \boldsymbol{\pi}_{\rm l}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

Let us define for all $P \in \mathcal{P}_{\rm S}$ the backward reachability set as follows:

$$brs(P) = \{P' \mid \exists a, \lambda.\, P' \xrightarrow{a,\lambda} P\}$$

and for all $P, P' \in \mathcal{P}_{\rm S}$ the backward rate as follows:

$$rate_{\rm b}(P', P) = \sum\{\!| \lambda \mid \exists a.\, P' \xrightarrow{a,\lambda} P |\!\}$$

As far as the transient case is concerned, we preliminarily recall that, given the infinitesimal generator matrix $\mathbf{Q} = [q_{h,l}]$ and the initial probability vector $\boldsymbol{\pi}_{\mathbf{Q}}^{(0)}$ of an arbitrary finite-state CTMC and chosen $q \in \mathbb{R}_{>0}$ such that $q \geq \max_{h,l} |q_{h,l}|$, the transient probability vector at time $t \in \mathbb{R}_{>0}$ is given by:

$$\boldsymbol{\pi}_{\mathbf{Q}}^{(t)} = \sum_{d=0}^{\infty} {\rm e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathbf{P}}^{(d)}$$

where:

$$\mathbf{P} = \frac{1}{q} \cdot \mathbf{Q} + \mathbf{I}$$

is the probability matrix of the DTMC embedded in the q-uniformized version of the given CTMC and:

$$\boldsymbol{\pi}_{\mathbf{P}}^{(d)} = \boldsymbol{\pi}_{\mathbf{P}}^{(d-1)} \cdot \mathbf{P} = \boldsymbol{\pi}_{\mathbf{P}}^{(d-2)} \cdot \mathbf{P}^2 = \ldots = \boldsymbol{\pi}_{\mathbf{P}}^{(0)} \cdot \mathbf{P}^d$$

is the transient probability vector of the DTMC after $d \in \mathbb{N}_{>0}$ steps, with $\boldsymbol{\pi}_{\mathbf{P}}^{(0)} = \boldsymbol{\pi}_{\mathbf{Q}}^{(0)}$.

Let us denote by $\boldsymbol{\pi}_{\mathrm{l}}^{(d)}$ (resp. $\boldsymbol{\pi}_{\mathrm{r}}^{(d)}$) the vector containing for each state of the DTMC embedded in the q-uniformized version of the original CTMC on the left (resp. the aggregated CTMC on the right) the probability of being in that state after d steps. As uniformization rate q we take the maximum absolute value occurring in the infinitesimal generator matrices of the two CTMCs.

We now proceed by induction on $d \in \mathbb{N}$ to prove the exact aggregation property for the two DTMCs in the transient case. From this fact, i.e. for all $d \in \mathbb{N}$:

$$\begin{aligned}
\boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s'] &= \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s_0] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s_i] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s] &= \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

it will follow that for all $t \in \mathbb{R}_{>0}$:

$$\begin{aligned}
\boldsymbol{\pi}_{\mathrm{r}}^{(t)}[s'] &= \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s'] = \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s_0] &&= \boldsymbol{\pi}_{\mathrm{l}}^{(t)}[s_0] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(t)}[s''] &= \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s''] = \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \sum_{i \in I} \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s_i] &&= \sum_{i \in I} \boldsymbol{\pi}_{\mathrm{l}}^{(t)}[s_i] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(t)}[s] &= \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathrm{r}}^{(d)}[s] = \sum_{d=0}^{\infty} \mathrm{e}^{-q \cdot t} \cdot \frac{(q \cdot t)^d}{d!} \cdot \boldsymbol{\pi}_{\mathrm{l}}^{(d)}[s] &&= \boldsymbol{\pi}_{\mathrm{l}}^{(t)}[s]
\end{aligned}$$

with s being as usual any other state not in $\{s', s''\}$.

If $d = 0$ then $\boldsymbol{\pi}_{\mathrm{l}}^{(d)}$ and $\boldsymbol{\pi}_{\mathrm{r}}^{(d)}$ trivially meet exactness because we assumed:

$$\begin{aligned}
\boldsymbol{\pi}_{\mathrm{r}}^{(0)}[s'] &= \boldsymbol{\pi}_{\mathrm{l}}^{(0)}[s_0] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(0)}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\mathrm{l}}^{(0)}[s_i] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(0)}[s] &= \boldsymbol{\pi}_{\mathrm{l}}^{(0)}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

Consider now $\boldsymbol{\pi}_{\mathrm{l}}^{(d)}$ and $\boldsymbol{\pi}_{\mathrm{r}}^{(d)}$ for $d \in \mathbb{N}_{>0}$ and suppose that exactness holds for all $d' \in \mathbb{N}_{<d}$, i.e.:

$$\begin{aligned}
\boldsymbol{\pi}_{\mathrm{r}}^{(d')}[s'] &= \boldsymbol{\pi}_{\mathrm{l}}^{(d')}[s_0] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(d')}[s''] &= \sum_{i \in I} \boldsymbol{\pi}_{\mathrm{l}}^{(d')}[s_i] \\
\boldsymbol{\pi}_{\mathrm{r}}^{(d')}[s] &= \boldsymbol{\pi}_{\mathrm{l}}^{(d')}[s] \qquad \text{any other state } s \notin \{s', s''\}
\end{aligned}$$

The linear equation system after d steps for the DTMC embedded in the q-uniformized version of the original CTMC on the left is the following:

$$\begin{aligned}
\pi_{\mathrm{l}}^{(d)}[s_0] &= \sum_{P' \in brs(s_0)} \pi_{\mathrm{l}}^{(d-1)}[P'] \cdot \tfrac{rate_{\mathrm{b}}(P',s_0)}{q} & \\
\pi_{\mathrm{l}}^{(d)}[s_i] &= \pi_{\mathrm{l}}^{(d-1)}[s_0] \cdot \tfrac{\lambda_i}{q} & i \in I \\
\pi_{\mathrm{l}}^{(d)}[P_{i,j}] &= \pi_{\mathrm{l}}^{(d-1)}[s_i] \cdot \tfrac{\mu_{i,j}}{q} & i \in I, j \in J_i \\
\pi_{\mathrm{l}}^{(d)}[P] &= \sum_{P' \in brs(P)} \pi_{\mathrm{l}}^{(d-1)}[P'] \cdot \tfrac{rate_{\mathrm{b}}(P',P)}{q} & \text{any other state } P
\end{aligned}$$

while for the DTMC embedded in the q-uniformized version of the aggregated CTMC on the right we have:

$$\begin{aligned}
\pi_{\mathrm{r}}^{(d)}[s'] &= \sum_{P' \in brs(s')} \pi_{\mathrm{r}}^{(d-1)}[P'] \cdot \tfrac{rate_{\mathrm{b}}(P',s')}{q} & \\
\pi_{\mathrm{r}}^{(d)}[s''] &= \pi_{\mathrm{r}}^{(d-1)}[s'] \cdot \sum_{k \in I} \tfrac{\lambda_k}{q} & \\
\pi_{\mathrm{r}}^{(d)}[P_{i,j}] &= \pi_{\mathrm{r}}^{(d-1)}[s''] \cdot \tfrac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \tfrac{\mu_{i,j}}{q} & i \in I, j \in J_i \\
\pi_{\mathrm{r}}^{(d)}[P] &= \sum_{P' \in brs(P)} \pi_{\mathrm{r}}^{(d-1)}[P'] \cdot \tfrac{rate_{\mathrm{b}}(P',P)}{q} & \text{any other state } P
\end{aligned}$$

with both the $\pi_{\mathrm{l}}^{(d)}[.]$'s and the $\pi_{\mathrm{r}}^{(d)}[.]$'s summing up to 1.
We now show that, through the introduction for all $d'' \in \mathbb{N}_{\leq d}$ of a new variable $y^{(d'')}$ replacing the set of variables $\{\pi_{\mathrm{l}}^{(d'')}[s_i] \mid i \in I\}$, the system of linear equations for the DTMC associated with the original CTMC on the left can be transformed into a linear system having the same number of variables and equations as well as the same coefficient matrix as the system of linear equations for the DTMC associated with the aggregated CTMC on the right.
By summing up over all $i \in I$ the second group of equations in the linear system for the DTMC associated with the original CTMC, we derive:

$$\sum_{i \in I} \pi_{\mathrm{l}}^{(d)}[s_i] = \pi_{\mathrm{l}}^{(d-1)}[s_0] \cdot \sum_{k \in I} \tfrac{\lambda_k}{q}$$

which can be rewritten as follows:

$$y^{(d)} = \pi_{\mathrm{l}}^{(d-1)}[s_0] \cdot \sum_{k \in I} \tfrac{\lambda_k}{q}$$

if we let:

$$y^{(d)} = \sum_{i \in I} \pi_{\mathrm{l}}^{(d)}[s_i]$$

Observed that:

$$\pi_{\mathrm{l}}^{(d-1)}[s_0] = y^{(d)} \cdot \tfrac{q}{\sum_{k \in I} \lambda_k}$$

hence for all $i \in I$:

$$\pi_{\mathrm{l}}^{(d)}[s_i] = y^{(d)} \cdot \tfrac{q}{\sum_{k \in I} \lambda_k} \cdot \tfrac{\lambda_i}{q} = y^{(d)} \cdot \tfrac{\lambda_i}{\sum_{k \in I} \lambda_k}$$

it will be:

$$\pi_{\mathrm{l}}^{(d-1)}[s_i] = y^{(d-1)} \cdot \tfrac{\lambda_i}{\sum_{k \in I} \lambda_k}$$

where:

$$y^{(d-1)} = \sum_{i \in I} \pi_{\mathrm{l}}^{(d-1)}[s_i]$$

Therefore the third group of equations in the linear system for the DTMC associated with the original CTMC can be rewritten as follows:

$$\pi_{\mathrm{l}}^{(d)}[P_{i,j}] = y^{(d-1)} \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \frac{\mu_{i,j}}{q} \qquad i \in I, j \in J_i$$

In conclusion, the introduction of the new variables $y^{(d'')}$'s causes the system of linear equations for the DTMC associated with the original CTMC to be equivalent to the following one:

$$\begin{aligned}
\pi_{\mathrm{l}}^{(d)}[s_0] &= \sum_{P' \in brs(s_0)} \pi_{\mathrm{l}}^{(d-1)}[P'] \cdot \frac{rate_{\mathrm{b}}(P', s_0)}{q} & \\
y^{(d)} &= \pi_{\mathrm{l}}^{(d-1)}[s_0] \cdot \sum_{k \in I} \frac{\lambda_k}{q} & \\
\pi_{\mathrm{l}}^{(d)}[P_{i,j}] &= y^{(d-1)} \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \frac{\mu_{i,j}}{q} & i \in I, j \in J_i \\
\pi_{\mathrm{l}}^{(d)}[P] &= \sum_{P' \in brs(P)} \pi_{\mathrm{l}}^{(d-1)}[P'] \cdot \frac{rate_{\mathrm{b}}(P', P)}{q} & \text{any other state } P
\end{aligned}$$

with all the occurring $\pi_{\mathrm{l}}^{(d)}[.]$'s plus $y^{(d)}$ summing up to 1, which has the same form as the system of linear equations for the DTMC associated with the aggregated CTMC. By virtue of the induction hypothesis we have:

$$\begin{aligned}
\pi_{\mathrm{l}}^{(d-1)}[P'] &= \pi_{\mathrm{r}}^{(d-1)}[P'] \qquad P' \in brs(s_0) = brs(s') \\
\pi_{\mathrm{l}}^{(d-1)}[s_0] &= \pi_{\mathrm{r}}^{(d-1)}[s'] \\
y^{(d-1)} &= \pi_{\mathrm{r}}^{(d-1)}[s''] \\
\pi_{\mathrm{l}}^{(d-1)}[P'] &= \pi_{\mathrm{r}}^{(d-1)}[P'] \qquad P' \in brs(P), \text{ any other state } P
\end{aligned}$$

hence:

$$\begin{aligned}
\pi_{\mathrm{l}}^{(d)}[s_0] &= \pi_{\mathrm{r}}^{(d)}[s'] \\
y^{(d)} &= \pi_{\mathrm{r}}^{(d)}[s''] \\
\pi_{\mathrm{l}}^{(d)}[P_{i,j}] &= \pi_{\mathrm{r}}^{(d)}[P_{i,j}] \qquad i \in I, j \in J_i \\
\pi_{\mathrm{l}}^{(d)}[P] &= \pi_{\mathrm{r}}^{(d)}[P] \qquad \text{any other state } P
\end{aligned}$$

from which exactness follows since $y^{(d)} = \sum_{i \in I} \pi_{\mathrm{l}}^{(d)}[s_i]$.

Whenever the two stationary probability vectors $\boldsymbol{\pi}_{\mathrm{l}}$ and $\boldsymbol{\pi}_{\mathrm{r}}$ exist, $\boldsymbol{\pi}_{\mathrm{l}}$ satisfies the following linear system of global balance equations for the original CTMC on the left:

$$\begin{aligned}
\pi_{\mathrm{l}}[s_i] \cdot \mu &= \pi_{\mathrm{l}}[s_0] \cdot \lambda_i & i \in I \\
\pi_{\mathrm{l}}[P_{i,j}] \cdot rate_{\mathrm{t}}(P_{i,j}, 0) &= \pi_{\mathrm{l}}[s_i] \cdot \mu_{i,j} & i \in I, j \in J_i \\
\pi_{\mathrm{l}}[P] \cdot rate_{\mathrm{t}}(P, 0) &= \sum_{P' \in brs(P)} \pi_{\mathrm{l}}[P'] \cdot rate_{\mathrm{b}}(P', P) & \text{any other state } P
\end{aligned}$$

while for the aggregated CTMC on the right we have that $\boldsymbol{\pi}_{\mathrm{r}}$ satisfies:

$$\begin{aligned}
\pi_{\mathrm{r}}[s''] \cdot \mu &= \pi_{\mathrm{r}}[s'] \cdot \sum_{k \in I} \lambda_k & \\
\pi_{\mathrm{r}}[P_{i,j}] \cdot rate_{\mathrm{t}}(P_{i,j}, 0) &= \pi_{\mathrm{r}}[s''] \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \mu_{i,j} & i \in I, j \in J_i \\
\pi_{\mathrm{r}}[P] \cdot rate_{\mathrm{t}}(P, 0) &= \sum_{P' \in brs(P)} \pi_{\mathrm{r}}[P'] \cdot rate_{\mathrm{b}}(P', P) & \text{any other state } P
\end{aligned}$$

with both the $\pi_{\rm l}[.]$'s and the $\pi_{\rm r}[.]$'s summing up to 1.
Similarly to the transient case, we now show that, through the introduction of a new variable y replacing the set of variables $\{\pi_{\rm l}[s_i] \mid i \in I\}$, the system of linear equations for the original CTMC on the left can be transformed into a linear system having the same number of variables and equations as well as the same coefficient matrix as the system of linear equations for the aggregated CTMC on the right.
By summing up over all $i \in I$ the first group of equations in the linear system for the original CTMC, we derive:

$$\sum_{i \in I} \pi_{\rm l}[s_i] \cdot \mu = \pi_{\rm l}[s_0] \cdot \sum_{k \in I} \lambda_k$$

which can be rewritten as follows:

$$y \cdot \mu = \pi_{\rm l}[s_0] \cdot \sum_{k \in I} \lambda_k$$

if we let:

$$y = \sum_{i \in I} \pi_{\rm l}[s_i]$$

Observed that:

$$\pi_{\rm l}[s_0] = y \cdot \frac{\mu}{\sum_{k \in I} \lambda_k}$$

hence for all $i \in I$:

$$\pi_{\rm l}[s_i] = \pi_{\rm l}[s_0] \cdot \frac{\lambda_i}{\mu} = y \cdot \frac{\mu}{\sum_{k \in I} \lambda_k} \cdot \frac{\lambda_i}{\mu} = y \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k}$$

the second group of equations in the linear system for the original CTMC can be rewritten as follows:

$$\pi_{\rm l}[P_{i,j}] \cdot rate_{\rm t}(P_{i,j}, 0) = y \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \mu_{i,j} \qquad i \in I, j \in J_i$$

In conclusion, the introduction of variable y causes the system of linear equations for the original CTMC to be equivalent to the following one:

$$\begin{aligned}
y \cdot \mu &= \pi_{\rm l}[s_0] \cdot \sum_{k \in I} \lambda_k \\
\pi_{\rm l}[P_{i,j}] \cdot rate_{\rm t}(P_{i,j}, 0) &= y \cdot \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \mu_{i,j} && i \in I, j \in J_i \\
\pi_{\rm l}[P] \cdot rate_{\rm t}(P, 0) &= \sum_{P' \in brs(P)} \pi_{\rm l}[P'] \cdot rate_{\rm b}(P', P) && \text{any other state } P
\end{aligned}$$

with all the occurring $\pi_{\rm l}[.]$'s plus y summing up to 1, which has the same form as the system of linear equations for the aggregated CTMC. As a consequence:

$$\begin{aligned}
\pi_{\rm l}[s_0] &= \pi_{\rm r}[s'] \\
y &= \pi_{\rm r}[s''] \\
\pi_{\rm l}[P_{i,j}] &= \pi_{\rm r}[P_{i,j}] && i \in I, j \in J_i \\
\pi_{\rm l}[P] &= \pi_{\rm r}[P] && \text{any other state } P
\end{aligned}$$

from which exactness follows since $y = \sum_{i \in I} \pi_{\rm l}[s_i]$. ■

5 Markovian Trace Equivalence for SMPC

In this section we introduce and investigate the properties of another non-bisimulation-based Markovian behavioral equivalence – originally defined in [23] – which relies on the probability of exhibiting certain computations that are performed within certain amounts of time. Like in the case of Markovian testing equivalence, for the time being we present this Markovian trace equivalence by restricting ourselves to SMPC.

5.1 Equivalence Definition

Unlike Markovian testing equivalence, given a process term $P \in \mathcal{P}_{\mathrm{S}}$ we no longer have tests that interact with P in the case of Markovian trace equivalence. Instead, we directly consider the multiset $\mathcal{C}_{\mathrm{f}}(P)$ of the finite-length computations of P taken in isolation.

Definition 5.1 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\alpha \in \mathit{Name}^*$. We say that c is compatible with α iff:

$$\mathit{trace}(c) = \alpha$$

We denote by $\mathcal{CC}(P, \alpha)$ the multiset of the finite-length computations of P that are compatible with α. ■

Note that $\mathcal{CC}(P, \alpha) \subseteq \mathcal{I}_{\mathrm{f}}(P)$, because of the compatibility of the computations with the same trace α, and that $\mathcal{CC}(P, \alpha)$ is finite, because of the finitely-branching structure of the considered terms.

Definition 5.2 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian trace equivalent to P_2, written $P_1 \sim_{\mathrm{MTr}} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

$$\mathit{prob}(\mathcal{CC}_{\leq\theta}(P_1, \alpha)) = \mathit{prob}(\mathcal{CC}_{\leq\theta}(P_2, \alpha)) \quad ■$$

We now provide some necessary conditions for $\sim_{\mathrm{MTr}}$, which are based on the average durations of computations exhibiting the same traces as well as on the total exit rates of the last configurations of such computations. These necessary conditions, some of which are looser than those shown in Sect. 4.2 for $\sim_{\mathrm{MT}}$, will be useful in the remainder of Sect. 5.

Proposition 5.3 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$ and $\alpha \in \mathit{Name}^*$. Whenever $P_1 \sim_{\mathrm{MTr}} P_2$, then for all $c_k \in \mathcal{CC}(P_k, \alpha)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{CC}(P_h, \alpha)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$\mathit{time}_{\mathrm{a}}(c_k) = \mathit{time}_{\mathrm{a}}(c_h)$$

and:

$$\mathit{rate}_{\mathrm{t}}(P_{k,\mathrm{last}}, 0) = \mathit{rate}_{\mathrm{t}}(P_{h,\mathrm{last}}, 0)$$

with $P_{k,\mathrm{last}}$ (resp. $P_{h,\mathrm{last}}$) being the last configuration of c_k (resp. c_h).

Proof Taken $c_k \in \mathcal{CC}(P_k, \alpha)$, we proceed by induction on $length(\alpha)$:

- If $length(\alpha) = 0$ then necessarily $\alpha \equiv \varepsilon$. As a consequence, we immediately derive that there exists $c_h \in \mathcal{CC}(P_h, \alpha)$ such that:
$$time_{\mathrm{a}}(c_k) = \varepsilon = time_{\mathrm{a}}(c_h)$$
with $P_{k,\mathrm{last}} \equiv P_k \sim_{\mathrm{MTr}} P_h \equiv P_{h,\mathrm{last}}$. Now suppose that $P_{k,\mathrm{last}}$ and $P_{h,\mathrm{last}}$ have a different total exit rate, say e.g.:
$$rate_{\mathrm{t}}(P_{k,\mathrm{last}}, 0) > rate_{\mathrm{t}}(P_{h,\mathrm{last}}, 0)$$
Then necessarily $P_{k,\mathrm{last}}$ and $P_{h,\mathrm{last}}$ have a different total exit rate with respect to some $a \in Name$:
$$rate(P_{k,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}}) > rate(P_{h,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}})$$
If we considered the trace $\alpha' \equiv a$, for $\theta = 1/rate_{\mathrm{t}}(P_{k,\mathrm{last}}, 0)$ we would have:
$$prob(\mathcal{CC}_{\leq\theta}(P_{k,\mathrm{last}}, \alpha')) > 0 = prob(\mathcal{CC}_{\leq\theta}(P_{h,\mathrm{last}}, \alpha'))$$
which contradicts $P_{k,\mathrm{last}} \sim_{\mathrm{MTr}} P_{h,\mathrm{last}}$. As a consequence, it must be:
$$rate(P_{k,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}}) = rate(P_{h,\mathrm{last}}, a, 0, \mathcal{P}_{\mathrm{C}})$$
for all $a \in Name$ and hence:
$$rate_{\mathrm{t}}(P_{k,\mathrm{last}}, 0) = rate_{\mathrm{t}}(P_{h,\mathrm{last}}, 0)$$
- Let $length(\alpha) = n > 0$, with $\alpha \equiv \alpha' \circ b$. Let $c'_k \in \mathcal{CC}(P_k, \alpha')$ be the contraction by one step of c_k. Since $length(\alpha') = n - 1$, by the induction hypothesis there exists $c'_h \in \mathcal{CC}(P_h, \alpha')$ such that:
$$time_{\mathrm{a}}(c'_k) = time_{\mathrm{a}}(c'_h)$$
and:
$$rate_{\mathrm{t}}(P'_{k,\mathrm{last}}, 0) = rate_{\mathrm{t}}(P'_{h,\mathrm{last}}, 0)$$
with $P'_{k,\mathrm{last}}$ (resp. $P'_{h,\mathrm{last}}$) being the last configuration of c'_k (resp. c'_h). As a consequence:
$$\begin{aligned} time_{\mathrm{a}}(c_k) &= time_{\mathrm{a}}(c'_k) \circ \tfrac{1}{rate_{\mathrm{t}}(P'_{k,\mathrm{last}}, 0)} = \\ &= time_{\mathrm{a}}(c'_h) \circ \tfrac{1}{rate_{\mathrm{t}}(P'_{h,\mathrm{last}}, 0)} = time_{\mathrm{a}}(c_h) \end{aligned}$$
where $c_h \in \mathcal{CC}(P_h, \alpha)$ is one of the extensions by one step of c'_h according to α, which must exist in order not to violate $P_k \sim_{\mathrm{MTr}} P_h$.
Now assume that for each such c_h:
$$rate_{\mathrm{t}}(P_{k,\mathrm{last}}, 0) \neq rate_{\mathrm{t}}(P_{h,\mathrm{last}}, 0)$$
where $P_{k,\mathrm{last}}$ (resp. $P_{h,\mathrm{last}}$) is the last configuration of c_k (resp. c_h). Then we can build a trace that distinguishes P_k from P_h.
In fact, let us call weak sort a set of computations that intersects $\mathcal{CC}(P_k, \alpha)$ and $\mathcal{CC}(P_h, \alpha)$ if it comprises all the computations with (the same trace and) the same average duration whose last configurations have the same total exit rate. Note that, due to $P_k \sim_{\mathrm{MTr}} P_h$, for all $a \in Name$ the probability of performing a computation of a weak sort in $\mathcal{CC}'(P_k, \alpha)$ extended with an a-transition is the same as the probability of performing a computation of the weak sort in $\mathcal{CC}'(P_h, \alpha)$ extended with an a-transition.
After removing from $\mathcal{CC}(P_k, \alpha)$ and $\mathcal{CC}(P_h, \alpha)$ every weak sort, at least one

of the two sets of remaining computations – which we denote by $\mathcal{CC}'(P_k, \alpha)$ and $\mathcal{CC}'(P_h, \alpha)$ – will be non-empty because of the assumption that c_k is not matched by any c_h deriving from the extension by one step of c'_h according to α. Then there exist some remaining computations with the same average duration and in the same set, say e.g. $\mathcal{CC}'(P_k, \alpha)$, such that the last configuration of each of them has the same maximum total exit rate $\bar{r}$, while each of the other remaining computations has a different average duration or its last configuration has a lower total exit rate. Denoted by a the name of one of the actions enabled in the last configurations of the considered remaining computations, if we took the trace $\alpha'' \equiv \alpha \circ a$, then for some suitable $\bar{\theta}$ such that $length(\bar{\theta}) = length(\alpha)$ we would have:

$$prob(\mathcal{CC}_{\leq \bar{\theta} \circ \frac{1}{\bar{r}}}(P_k, \alpha'')) = p_k + q_k > q_h = prob(\mathcal{CC}_{\leq \bar{\theta} \circ \frac{1}{\bar{r}}}(P_h, \alpha''))$$

where $q_k, q_h \geq 0$ are the possible contributions of weak sorts (whose average duration does not exceed $\bar{\theta}$ and whose last configurations have a total exit rate that does not exceed $\bar{r}$), with $q_k = q_h$ by virtue of $P_k \sim_{\rm MTr} P_h$. Since the above inequality contradicts $P_k \sim_{\rm MTr} P_h$, for at least one c_h deriving from the extension by one step of c'_h according to α it must be:

$$rate_{\rm t}(P_{k,{\rm last}}, 0) = rate_{\rm t}(P_{h,{\rm last}}, 0) \qquad \blacksquare$$

Corollary 5.4 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Whenever $P_1 \sim_{\rm MTr} P_2$, then for all $c_k \in \mathcal{C}_{\rm f}(P_k)$ with $k \in \{1,2\}$ there exists $c_h \in \mathcal{C}_{\rm f}(P_h)$ with $h \in \{1,2\} - \{k\}$ such that:

$$trace(c_k) = trace(c_h) \ \wedge \ time_{\rm a}(c_k) = time_{\rm a}(c_h) \qquad \blacksquare$$

Corollary 5.5 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Whenever $P_1 \sim_{\rm MTr} P_2$, then for all $a \in Name$:

$$rate(P_1, a, 0, \mathcal{P}_{\rm C}) = rate(P_2, a, 0, \mathcal{P}_{\rm C}) \qquad \blacksquare$$

Corollary 5.6 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Whenever $P_1 \sim_{\rm MTr} P_2$, then:

$$rate_{\rm t}(P_1, 0) = rate_{\rm t}(P_2, 0) \qquad \blacksquare$$

5.2 Alternative Characterizations

We now provide the following alternative characterizations of Markovian trace equivalence:

(1) The first characterization is based on a predicate establishing whether a process term is capable of executing a trace with a probability that is above a given threshold, by taking an average time that is below another given threshold. This characterization will be useful in Sect. 5.3 to establish connections between $\sim_{\rm MTr}$ and its nondeterministic and probabilistic counterparts.

(2) The second characterization is based on the probability distribution of executing a trace within a certain average amount of time. This characterization, which results in a slightly different definition with respect to the one of $\sim_{\mathrm{MTr}}$, will be useful to prove the third characterization.
(3) The third characterization is based on the probability distribution of executing a trace within a certain amount of time. This characterization, which involves random variables instead of their expected values, justifies the definition of $\sim_{\mathrm{MTr}}$ in terms of the average durations of the computations, which are easier to work with than the probability distributions quantifying the same durations. This result was originally proved in [23] with a different technique.

5.2.1 First Characterization: Execute Predicate

The first alternative characterization of $\sim_{\mathrm{MTr}}$ relies on a predicate establishing whether a process term is capable of executing a trace with a probability that is above a given threshold, by taking an average time that is below another given threshold.

Definition 5.7 Let $P \in \mathcal{P}_{\mathrm{S}}$, $\alpha \in \mathit{Name}^*$, $p \in \mathbb{R}_{]0,1]}$, and $\theta \in \mathbb{R}^*_{>0}$. We say that P executes α with probability at least p within a sequence θ of time units on average, written $P\ \mathit{execute}_{p,\theta}\ \alpha$, iff:

$$\mathit{prob}(\mathcal{CC}_{\leq\theta}(P, \alpha)) \geq p$$

■

Definition 5.8 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian execute-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MTr,execute}} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$, probabilities $p \in \mathbb{R}_{]0,1]}$, and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

$$P_1\ \mathit{execute}_{p,\theta}\ \alpha \iff P_2\ \mathit{execute}_{p,\theta}\ \alpha$$

■

Proposition 5.9 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MTr,execute}} P_2 \iff P_1 \sim_{\mathrm{MTr}} P_2$$

Proof Similar to the proof of Prop. 4.12. ■

5.2.2 Second Characterization: Average Durations

The second alternative characterization of $\sim_{\mathrm{MTr}}$ is based on the probability distribution of executing a trace within a certain average amount of time.

Definition 5.10 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian average-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MTr,a}} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of average amounts of time:

$$\mathit{prob}_{\mathrm{a}}(\mathcal{CC}(P_1, \alpha), \theta) = \mathit{prob}_{\mathrm{a}}(\mathcal{CC}(P_2, \alpha), \theta)$$

■

Proposition 5.11 Let $P_1, P_2 \in \mathcal{P}_S$. Then:

$$P_1 \sim_{\text{MTr,a}} P_2 \iff P_1 \sim_{\text{MTr}} P_2$$

Proof A straightforward consequence of Lemma 4.14. ■

5.2.3 Third Characterization: Duration Distributions

The third alternative characterization of $\sim_{\text{MTr}}$ is based on the probability distribution of executing a trace within a certain amount of time. A consequence of this result is that considering the (more accurate) probability distributions quantifying the durations of the computations leads to the same equivalence as considering the (easier to work with) average durations of the computations, hence to $\sim_{\text{MTr}}$.

Definition 5.12 Let $P_1, P_2 \in \mathcal{P}_S$. We say that P_1 is Markovian distribution-trace equivalent to P_2, written $P_1 \sim_{\text{MTr,d}} P_2$, iff for all traces $\alpha \in Name^*$ and sequences $\theta \in \mathbb{R}^*_{>0}$ of amounts of time:

$$prob_{\text{d}}(\mathcal{CC}(P_1, \alpha), \theta) = prob_{\text{d}}(\mathcal{CC}(P_2, \alpha), \theta)$$ ■

Lemma 5.13 Let $P_1, P_2 \in \mathcal{P}_S$ and $\alpha \in Name^*$. Whenever $P_1 \sim_{\text{MTr,d}} P_2$, then for all $c_k \in \mathcal{CC}(P_k, \alpha)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{CC}(P_h, \alpha)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$time_{\text{d}}(c_k) = time_{\text{d}}(c_h)$$

and:

$$rate_{\text{t}}(P_{k,\text{last}}, 0) = rate_{\text{t}}(P_{h,\text{last}}, 0)$$

with $P_{k,\text{last}}$ (resp. $P_{h,\text{last}}$) being the last configuration of c_k (resp. c_h).

Proof Similar to the proof of Prop. 5.3. ■

Corollary 5.14 Let $P_1, P_2 \in \mathcal{P}_S$. Whenever $P_1 \sim_{\text{MTr}} P_2$, then for all $c_k \in \mathcal{C}_f(P_k)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{C}_f(P_h)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$trace(c_k) = trace(c_h) \ \wedge \ time_{\text{d}}(c_k) = time_{\text{d}}(c_h)$$ ■

Theorem 5.15 Let $P_1, P_2 \in \mathcal{P}_S$. Then:

$$P_1 \sim_{\text{MTr,d}} P_2 \iff P_1 \sim_{\text{MTr,a}} P_2$$

Proof Similar to the proof of Thm. 4.20, with the difference that $\mathcal{CC}(P_k, \alpha)$ has to be considered instead of $\mathcal{SC}(P_k, T)$ for each $k = 1, 2$, and that Cor. 5.14 and Lemma 4.19 – for $\Longrightarrow$ – and Cor. 5.4 (via Prop. 5.11) – for $\Longleftarrow$ – have to be used. ■

Corollary 5.16 Let $P_1, P_2 \in \mathcal{P}_S$. Then:

$$P_1 \sim_{\text{MTr,d}} P_2 \iff P_1 \sim_{\text{MTr}} P_2$$ ■

We now prove the following properties of Markovian trace equivalence:

(1) $\sim_{\mathrm{MTr}}$ is a strict refinement of nondeterministic trace equivalence [17].
(2) $\sim_{\mathrm{MTr}}$ is a strict refinement of probabilistic trace equivalence [18].
(3) $\sim_{\mathrm{MTr}}$ is strictly coarser than $\sim_{\mathrm{MT}}$.
(4) $\sim_{\mathrm{MTr}}$ and $\sim_{\mathrm{MT}}$ have precise connections with some variants of $\sim_{\mathrm{MTr}}$. From [23] it is known that $\sim_{\mathrm{MTr}}$ coincides with Markovian completed-trace equivalence – hence $\sim_{\mathrm{MTr}}$ is deadlock sensitive – and that Markovian failure-trace equivalence coincides with Markovian ready-trace equivalence. It can also be derived that Markovian failure equivalence coincides with Markovian ready equivalence from a similar result proved in [18] for the failure and ready variants of probabilistic trace equivalence.
Here we demonstrate that $\sim_{\mathrm{MT}}$ coincides with Markovian ready equivalence. This result, together with those mentioned above and those proved in Sect. 4.4 and Sect. 4.3.4, provides useful information about the Markovian linear-time/branching-time spectrum.

Definition 5.17 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $\alpha \in \mathit{Name}^*$. We say that P executes α, written $P\ \mathit{execute}\ \alpha$, iff at least one computation is compatible with α:

$$\mathcal{CC}(P, \alpha) \neq \emptyset$$

■

Definition 5.18 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is trace equivalent to P_2, written $P_1 \sim_{\mathrm{Tr}} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$:

$$P_1\ \mathit{execute}\ \alpha \iff P_2\ \mathit{execute}\ \alpha$$

■

Lemma 5.19 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $\alpha \in \mathit{Name}^*$. Then $P\ \mathit{execute}\ \alpha$ iff there exist $p \in \mathbb{R}_{]0,1]}$ and $\theta \in \mathbb{R}^*_{>0}$ such that $P\ \mathit{execute}_{p,\theta}\ \alpha$.

Proof Similar to the proof of Lemma 4.39(1). ■

Proposition 5.20 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MTr}} P_2 \implies P_1 \sim_{\mathrm{Tr}} P_2$$

Proof Similar to the first part of the proof of Prop. 4.40, with the difference that Lemma 5.19 and Prop. 5.9 have to be used. ■

Example 5.21 The converse of Prop. 5.20 does not hold, i.e. nondeterministic trace equivalence is strictly coarser than Markovian trace equivalence.
If we consider the same two process terms as Ex. 4.41:

$$P_1 \equiv <a, \lambda>.\underline{0} + <b, \mu>.\underline{0}$$

$$P_2 \equiv <a, \mu>.\underline{0} + <b, \lambda>.\underline{0}$$

with $a \neq b$ and $\lambda \neq \mu$, then $P_1 \sim_{\rm Tr} P_2$ because both process terms can only perform an a-action and a b-action, but $P_1 \not\sim_{\rm MTr} P_2$. For instance, the two process terms are distinguished by trace:

$$\alpha \equiv a$$

In fact, although P_1 and P_2 have the same average sojourn time $t = 1/(\lambda+\mu)$, the probability of executing α within t time units on average is $\lambda/(\lambda+\mu)$ in the case of P_1, while it is $\mu/(\lambda+\mu)$ in the case of P_2. ■

Definition 5.22 Let $P \in \mathcal{P}_{\rm S}$, $\alpha \in \mathit{Name}^*$, and $p \in \mathbb{R}_{]0,1]}$. We say that P executes α with probability at least p, written $P\ \mathit{execute}_p\ \alpha$, iff:

$$\mathit{prob}(\mathcal{CC}(P, \alpha)) \geq p$$ ■

Definition 5.23 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. We say that P_1 is probabilistic trace equivalent to P_2, written $P_1 \sim_{\rm PTr} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$ and probabilities $p \in \mathbb{R}_{]0,1]}$:

$$P_1\ \mathit{execute}_p\ \alpha \iff P_2\ \mathit{execute}_p\ \alpha$$ ■

Lemma 5.24 Let $P \in \mathcal{P}_{\rm S}$, $\alpha \in \mathit{Name}^*$, and $p \in \mathbb{R}_{]0,1]}$. Then $P\ \mathit{execute}_p\ \alpha$ iff there exists $\theta \in \mathbb{R}^*_{>0}$ such that $P\ \mathit{execute}_{p,\theta}\ \alpha$.

Proof Similar to the proof of Prop. 4.44. ■

Proposition 5.25 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Then:

$$P_1 \sim_{\rm MTr} P_2 \implies P_1 \sim_{\rm PTr} P_2$$

Proof Similar to the proof of Prop. 4.45, with the difference that Lemma 5.24 and Prop. 5.9 have to be used. ■

Example 5.26 The converse of Prop. 5.25 does not hold, i.e. probabilistic trace equivalence is strictly coarser than Markovian trace equivalence.
If we consider the same two process terms as Ex. 4.46:

$$P_3 \equiv <a, \lambda>.\underline{0} + <b, \mu>.\underline{0}$$

$$P_4 \equiv <a, 2 \cdot \lambda>.\underline{0} + <b, 2 \cdot \mu>.\underline{0}$$

with $a \neq b$, then $P_3 \sim_{\rm PTr} P_4$ because both process terms can only perform an a-action and a b-action with the same probabilities – $\lambda/(\lambda+\mu)$ and $\mu/(\lambda+\mu)$, respectively – but $P_3 \not\sim_{\rm MTr} P_4$. For instance, the two process terms are distinguished by trace:

$$\alpha \equiv a$$

In fact, the average sojourn time of P_3 is $1/(\lambda + \mu)$ whereas the average sojourn time of P_4 is $1/(2 \cdot \lambda + 2 \cdot \mu)$, hence the probability of executing α within $1/(2 \cdot \lambda + 2 \cdot \mu)$ time units on average is 0 in the case of P_3, while it is $\lambda/(\lambda + \mu)$ in the case of P_4. ■

Corollary 5.27 $\sim_{\rm MTr} \subset \sim_{\rm PTr} \subset \sim_{\rm Tr}$. ■

Definition 5.28 Let $\alpha \in \mathit{Name}^*$. The extended trace associated with α is defined by induction on the length of α through the following $\mathcal{ET}$-valued function:

$$\boxed{\mathit{trace}_{\rm e}(\alpha) = \begin{cases} \varepsilon & \text{if } \mathit{length}(\alpha) = 0 \\ (a, \mathit{Name}) \circ \mathit{trace}_{\rm e}(\alpha') & \text{if } \alpha \equiv a \circ \alpha' \end{cases}}$$

where ε is the empty extended trace. ■

Lemma 5.29 Let $P \in \mathcal{P}_{\rm S}$ and $\alpha \in \mathit{Name}^*$. Then for all $\theta \in \mathbb{R}^*_{>0}$:

$$\mathit{prob}^{\mathit{trace}_{\rm e}(\alpha)}(\mathcal{CC}^{\mathit{trace}_{\rm e}(\alpha)}_{\leq \theta}(P, \mathit{trace}_{\rm e}(\alpha))) = \mathit{prob}(\mathcal{CC}_{\leq \theta}(P, \alpha))$$

Proof A straightforward consequence of the fact that, due to Def. 5.28, $\mathcal{CC}(P, \mathit{trace}_{\rm e}(\alpha)) = \mathcal{CC}(P, \alpha)$ and for all $c \in \mathcal{CC}(P, \mathit{trace}_{\rm e}(\alpha))$ Def. 4.25 reduces to Def. 2.6 and Def. 4.26 reduces to Def. 2.7. ■

Proposition 5.30 Let $P_1, P_2 \in \mathcal{P}_{\rm S}$. Then:

$$P_1 \sim_{\rm MT} P_2 \implies P_1 \sim_{\rm MTr} P_2$$

Proof By virtue of Thm. 4.35, $P_1 \sim_{\rm MT} P_2$ means that for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^*_{>0}$:

$$\mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_1, \sigma)) = \mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_2, \sigma))$$

hence in particular for all $\alpha \in \mathit{Name}^*$ and $\theta \in \mathbb{R}^*_{>0}$:

$$\mathit{prob}^{\mathit{trace}_{\rm e}(\alpha)}(\mathcal{CC}^{\mathit{trace}_{\rm e}(\alpha)}_{\leq \theta}(P_1, \mathit{trace}_{\rm e}(\alpha))) = \mathit{prob}^{\mathit{trace}_{\rm e}(\alpha)}(\mathcal{CC}^{\mathit{trace}_{\rm e}(\alpha)}_{\leq \theta}(P_2, \mathit{trace}_{\rm e}(\alpha)))$$

The result then follows from Lemma 5.29. ■

Example 5.31 The converse of Prop. 5.30 does not hold, i.e. Markovian trace equivalence is strictly coarser than Markovian testing equivalence.
If we consider a slight variant of the two process terms of Ex. 4.49:

$$P_7 \equiv \,<\!a, \lambda_1\!>.<\!b, \mu\!>.P' + <\!a, \lambda_2\!>.<\!c, \mu\!>.P''$$

$$P_8 \equiv \,<\!a, \lambda_1 + \lambda_2\!>.(<\!b, \tfrac{\lambda_1}{\lambda_1 + \lambda_2} \cdot \mu\!>.P' + <\!c, \tfrac{\lambda_2}{\lambda_1 + \lambda_2} \cdot \mu\!>.P'')$$

then we have $P_7 \sim_{\rm MTr} P_8$ but $P_7 \not\sim_{\rm MT} P_8$ if $b \neq c$. In fact, no trace starting with a possibly followed by b or c can distinguish between the two terms, because in both terms the average time to perform an a-action followed by a b-action or a c-action is $1/(\lambda_1 + \lambda_2) \circ 1/\mu$ and the probability of reaching P' (resp. P'') is $\lambda_1/(\lambda_1 + \lambda_2)$ (resp. $\lambda_2/(\lambda_1 + \lambda_2)$). By contrast, any test starting with a passive a-action followed by either a passive b-action or a passive c-action can distinguish between the two terms as it increases the average sojourn time of the configuration involving $<b, \lambda_1/(\lambda_1 + \lambda_2) \cdot \mu>.P' + <c, \lambda_2/(\lambda_1 + \lambda_2) \cdot \mu>.P''$ from $1/\mu$ to $(\lambda_1 + \lambda_2)/\lambda_1 \cdot 1/\mu$ or $(\lambda_1 + \lambda_2)/\lambda_2 \cdot 1/\mu$, respectively. ■

To conclude, we introduce five variants of $\sim_{\mathrm{MTr}}$ based on the notions of completed trace, failure set, ready set, failure trace, and ready trace. We recall that a completed trace is a trace that ends up in a deadlock state, a failure set is a set of names of actions that cannot be executed in a certain state, a ready set is the set of the names of all the actions that must be executable in a certain state, a failure trace is a trace extended at each step with a failure set, and a ready trace is a trace extended at each step with a ready set.

Definition 5.32 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\alpha \in \mathit{Name}^*$. We say that c is a maximal computation compatible with α iff $c \in \mathcal{CC}(P, \alpha)$ and the last configuration of c is deadlocked. We denote by $\mathcal{MCC}(P, \alpha)$ the multiset of the finite-length maximal computations of P that are compatible with α. ■

Definition 5.33 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian completed-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MTr,c}} P_2$, iff for all traces $\alpha \in \mathit{Name}^*$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$\mathit{prob}(\mathcal{CC}_{\leq\theta}(P_1, \alpha)) = \mathit{prob}(\mathcal{CC}_{\leq\theta}(P_2, \alpha))$$
$$\mathit{prob}(\mathcal{MCC}_{\leq\theta}(P_1, \alpha)) = \mathit{prob}(\mathcal{MCC}_{\leq\theta}(P_2, \alpha))$$
■

Definition 5.34 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\varphi \equiv (\alpha, \mathcal{F}) \in \mathit{Name}^* \times 2^{\mathit{Name}}$. We say that c is a failure computation compatible with φ iff $c \in \mathcal{CC}(P, \alpha)$ and the last configuration of c cannot execute any action whose name belongs to the failure set $\mathcal{F}$. We denote by $\mathcal{FCC}(P, \varphi)$ the multiset of the finite-length failure computations of P that are compatible with φ. ■

Definition 5.35 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian failure equivalent to P_2, written $P_1 \sim_{\mathrm{MF}} P_2$, iff for all traces with final failure set $\varphi \in \mathit{Name}^* \times 2^{\mathit{Name}}$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$\mathit{prob}(\mathcal{FCC}_{\leq\theta}(P_1, \varphi)) = \mathit{prob}(\mathcal{FCC}_{\leq\theta}(P_2, \varphi))$$
■

Definition 5.36 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\rho \equiv (\alpha, \mathcal{R}) \in \mathit{Name}^* \times 2^{\mathit{Name}}$. We say that c is a ready computation compatible with ρ iff $c \in \mathcal{CC}(P, \alpha)$ and the set of the names of all the actions executable by the last configuration of c coincides with the ready set $\mathcal{R}$. We denote by $\mathcal{RCC}(P, \rho)$ the multiset of the finite-length ready computations of P that are compatible with ρ. ■

Definition 5.37 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian ready equivalent to P_2, written $P_1 \sim_{\mathrm{MR}} P_2$, iff for all traces with final ready set $\rho \in \mathit{Name}^* \times 2^{\mathit{Name}}$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$\mathit{prob}(\mathcal{RCC}_{\leq\theta}(P_1, \rho)) = \mathit{prob}(\mathcal{RCC}_{\leq\theta}(P_2, \rho))$$
■

Definition 5.38 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\phi \in (\mathit{Name} \times 2^{\mathit{Name}})^*$. We say that c is a failure-trace computation compatible with ϕ iff c is compatible with the trace component of ϕ and each configuration of c cannot execute any action whose name belongs to the corresponding failure set in the failure component of ϕ. We denote by $\mathcal{FTCC}(P, \phi)$ the multiset of the finite-length failure-trace computations of P that are compatible with ϕ. ■

Definition 5.39 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian failure-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MFTr}} P_2$, iff for all failure traces $\phi \in (\mathit{Name} \times 2^{\mathit{Name}})^*$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$\mathit{prob}(\mathcal{FTCC}_{\leq \theta}(P_1, \phi)) = \mathit{prob}(\mathcal{FTCC}_{\leq \theta}(P_2, \phi))$$ ■

Definition 5.40 Let $P \in \mathcal{P}_{\mathrm{C}}$, $c \in \mathcal{C}_{\mathrm{f}}(P)$, and $\varrho \in (\mathit{Name} \times 2^{\mathit{Name}})^*$. We say that c is a ready-trace computation compatible with ϱ iff c is compatible with the trace component of ϱ and the sets of the names of all the actions executable by the configurations of c coincide with the corresponding ready sets in the ready component of ϱ. We denote by $\mathcal{RTCC}(P, \varrho)$ the multiset of the finite-length ready-trace computations of P that are compatible with ϱ. ■

Definition 5.41 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. We say that P_1 is Markovian ready-trace equivalent to P_2, written $P_1 \sim_{\mathrm{MRTr}} P_2$, iff for all ready traces $\varrho \in (\mathit{Name} \times 2^{\mathit{Name}})^*$ and sequences $\theta \in \mathbb{R}_{>0}^*$ of average amounts of time:

$$\mathit{prob}(\mathcal{RTCC}_{\leq \theta}(P_1, \varrho)) = \mathit{prob}(\mathcal{RTCC}_{\leq \theta}(P_2, \varrho))$$ ■

Proposition 5.42 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Then:

$$P_1 \sim_{\mathrm{MR}} P_2 \iff P_1 \sim_{\mathrm{MT}} P_2$$

Proof ($\Longrightarrow$) We prove the contrapositive, so assume that $P_1 \not\sim_{\mathrm{MT}} P_2$. Then by virtue of Thm. 4.35 there exist $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}_{>0}^*$ such that:

$$\mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_1, \sigma)) \neq \mathit{prob}^{\sigma}(\mathcal{CC}^{\sigma}_{\leq \theta}(P_2, \sigma))$$

Let σ have minimal length among all the extended traces satisfying the above inequality, with $\sigma \equiv \sigma' \circ (a, \mathcal{E})$ and $\mathit{trace}(\sigma) = \alpha' \circ a$. Due to the minimality of the length of σ, for all $\theta' \in \mathbb{R}_{>0}^*$ we have:

$$\mathit{prob}^{\mathit{trace}_e(\alpha')}(\mathcal{CC}^{\mathit{trace}_e(\alpha')}_{\leq \theta'}(P_1, \mathit{trace}_e(\alpha'))) = \mathit{prob}^{\mathit{trace}_e(\alpha')}(\mathcal{CC}^{\mathit{trace}_e(\alpha')}_{\leq \theta'}(P_2, \mathit{trace}_e(\alpha')))$$

hence by virtue of Lemma 5.29:

$$\mathit{prob}(\mathcal{CC}_{\leq \theta'}(P_1, \alpha')) = \mathit{prob}(\mathcal{CC}_{\leq \theta'}(P_2, \alpha'))$$

There are two cases. If the set of the α'-derivatives of P_1 and the set of the α'-derivatives of P_2 result in the same family of ready sets, then from the initial inequality we derive that for some $\theta'' \in \mathbb{R}_{>0}^*$ lexicographically greater than θ:

$$\mathit{prob}(\mathcal{CC}_{\leq \theta''}(P_1, \alpha' \circ a)) \neq \mathit{prob}(\mathcal{CC}_{\leq \theta''}(P_2, \alpha' \circ a))$$

hence $P_1 \not\sim_{\mathrm{MTr}} P_2$, from which $P_1 \not\sim_{\mathrm{MR}} P_2$ follows.

If instead the set of the α'-derivatives of P_1 and the set of the α'-derivatives of P_2 result in different families of ready sets, then there is at least one ready set $\mathcal{R}$ possessed by some α'-derivatives of e.g. P_1 that is not possessed by any α'-derivative of P_2. Then for some $\theta'' \in \mathbb{R}^*_{>0}$ lexicographically greater than θ we have:

$$\mathit{prob}(\mathcal{RCC}_{\leq\theta''}(P_1, (\alpha', \mathcal{R}))) > 0 = \mathit{prob}(\mathcal{RCC}_{\leq\theta''}(P_2, (\alpha', \mathcal{R})))$$

hence $P_1 \not\sim_{\rm MR} P_2$.
($\Longleftarrow$) By virtue of Prop. 4.6, from $P_1 \sim_{\rm MT} P_2$ it follows that, given an arbitrary $T \in \mathcal{T}$ and an arbitrary $c \in \mathcal{C}_{\rm f}(T, {\rm s})$, for all $c_k \in \mathcal{ESC}(P_k, T, c)$ with $k \in \{1, 2\}$ there exists $c_h \in \mathcal{ESC}(P_h, T, c)$ with $h \in \{1, 2\} - \{k\}$ such that:

$$\mathit{trace}_{\rm a}(c_k) = \mathit{trace}_{\rm a}(c_h) \;\wedge\; \mathit{time}_{\rm a}(c_k) = \mathit{time}_{\rm a}(c_h)$$

and for all $a \in \mathit{Name}$:

$$\mathit{rate}(P_{k,\rm last}, a, 0, \mathcal{P}_{\rm C}) = \mathit{rate}(P_{h,\rm last}, a, 0, \mathcal{P}_{\rm C})$$

with $P_{k,\rm last}$ (resp. $P_{h,\rm last}$) being the process component of the last configuration of c_k (resp. c_h). As a consequence, given an arbitrary $\alpha \in \mathit{Name}^*$, for all $c_k \in \mathcal{CC}(P_k, \alpha)$ there exists $c_h \in \mathcal{CC}(P_h, \alpha)$ such that:

$$\mathit{time}_{\rm a}(c_k) = \mathit{time}_{\rm a}(c_h)$$

and for all $a \in \mathit{Name}$:

$$\mathit{rate}(P_{k,\rm last}, a, 0, \mathcal{P}_{\rm C}) = \mathit{rate}(P_{h,\rm last}, a, 0, \mathcal{P}_{\rm C})$$

In other words, any finite-length computation of one of the two process terms is matched by a finite-length computation of the other process term having the same trace, the same average duration, and the same ready set.
Let us consider the computations of $\mathcal{CC}(P_k, \alpha)$ and $\mathcal{CC}(P_h, \alpha)$ in order of non-decreasing extended average duration, which is given by their average duration concatenated with the inverse of the total exit rate of their last configuration. This results in a partition of $\mathcal{CC}(P_k, \alpha) \cup \mathcal{CC}(P_h, \alpha)$ such that each class collects all the groups of matching computations with the same extended average duration. Let $\bar{\theta}_1 <_{\rm lex} \bar{\theta}_2 <_{\rm lex} \ldots <_{\rm lex} \bar{\theta}_n$ – with $n \in \mathbb{N}_{>0}$ and $<_{\rm lex}$ being the lexicographical order – be the resulting extended average durations.
We now examine the class whose associated extended average duration is $\bar{\theta}_1$. Let us denote by $\mathcal{R}_{1,1}, \mathcal{R}_{1,2}, \ldots, \mathcal{R}_{1,m_1}$ the ready sets – in order of non-decreasing size – characterizing the groups of matching computations in the class being examined. As far as the group characterized by $\mathcal{R}_{1,1}$ is concerned, take a test $T_{1,1}$ starting with a sequence of passive actions whose names are those occurring in α, followed by $\sum_{a \in \mathcal{R}_{1,1}} <a, *_1>.{\rm s}$. Then from $P_k \sim_{\rm MT} P_h$ we derive:

$$\mathit{prob}(\mathcal{SC}_{\leq\bar{\theta}_1}(P_k, T_{1,1})) = \mathit{prob}(\mathcal{SC}_{\leq\bar{\theta}_1}(P_h, T_{1,1}))$$

where:

$$\mathit{prob}(\mathcal{SC}_{\leq\bar{\theta}_1}(P_k, T_{1,1})) = \mathit{prob}(\mathcal{RCC}_{\leq\bar{\theta}_1}(P_k, (\alpha, \mathcal{R}_{1,1})))$$
$$\mathit{prob}(\mathcal{SC}_{\leq\bar{\theta}_1}(P_h, T_{1,1})) = \mathit{prob}(\mathcal{RCC}_{\leq\bar{\theta}_1}(P_h, (\alpha, \mathcal{R}_{1,1})))$$

For the generic group of matching computations characterized by ready set

$\mathcal{R}_{1,j}$, with $2 \le j \le m_1$, we build a test $T_{1,j}$ starting with a sequence of passive actions whose names are those occurring in α, followed by $\sum_{a \in \mathcal{R}_{1,j}} <a, *_1>.\mathrm{s}$. Then from $P_k \sim_{\mathrm{MT}} P_h$ we derive:

$$prob(\mathcal{SC}_{\le \bar{\theta}_1}(P_k, T_{1,j})) = prob(\mathcal{SC}_{\le \bar{\theta}_1}(P_h, T_{1,j}))$$

where:

$$prob(\mathcal{SC}_{\le \bar{\theta}_1}(P_k, T_{1,j})) = prob(\mathcal{RCC}_{\le \bar{\theta}_1}(P_k, (\alpha, \mathcal{R}_{1,j}))) + q_{k,1,<j}$$
$$prob(\mathcal{SC}_{\le \bar{\theta}_1}(P_h, T_{1,j})) = prob(\mathcal{RCC}_{\le \bar{\theta}_1}(P_h, (\alpha, \mathcal{R}_{1,j}))) + q_{h,1,<j}$$

with $q_{k,1,<j} = q_{h,1,<j} \in \mathbb{R}_{[0,1[}$ being the possible contribution of the groups of matching computations characterized by ready sets $\mathcal{R}_{1,1}, \mathcal{R}_{1,2}, \ldots, \mathcal{R}_{1,j-1}$ (the contribution is present whenever at least one among $\mathcal{R}_{1,1}, \mathcal{R}_{1,2}, \ldots, \mathcal{R}_{1,j-1}$ is contained in $\mathcal{R}_{1,j}$).
For the generic class whose associated average duration is $\bar{\theta}_i$, with $2 \le i \le n$, we can reason in a similar way. The only difference is that, in the equalities relating the probability of $\mathcal{SC}$-like sets and the probability of $\mathcal{RCC}$-like sets, we have to take into account the possible additional contributions $q_{k,<i} = q_{h,<i} \in \mathbb{R}_{[0,1[}$ of the classes whose associated extended average durations are $\bar{\theta}_1, \bar{\theta}_2, \ldots, \bar{\theta}_{i-1}$ (the contribution can be present whenever at least one among the ready sets characterizing the groups of matching computations of the previously examined classes intersects the ready set under consideration).
Due to the generality of α, we can conclude that $P_1 \sim_{\mathrm{MR}} P_2$. ■

Corollary 5.43 The Markovian linear-time/branching-time spectrum is:

$$\sim_{\mathrm{MB}} \subset \sim_{\mathrm{MRTr}} = \sim_{\mathrm{MFTr}} \subset \sim_{\mathrm{MR}} = \sim_{\mathrm{MT}} = \sim_{\mathrm{MTr,e}} = \sim_{\mathrm{MF}} \subset \sim_{\mathrm{MTr,c}} = \sim_{\mathrm{MTr}}$$
■

5.4 Congruence Property

We now show that Markovian trace equivalence turns out to be a congruence with respect to all the operators of SMPC.

Lemma 5.44 Let $P \in \mathcal{P}_{\mathrm{S}}$ and $\alpha \in \mathit{Name}^*$. Then for all $<a, \lambda> \in \mathit{Act}_{\mathrm{S}}$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{CC}_{\le \theta}(P, \alpha)) = prob(\mathcal{CC}_{\le \frac{1}{\lambda} \circ \theta}(<a, \lambda>.P, a \circ \alpha))$$

Proof Similar to the proof of Lemma 4.50, with the difference that $\mathcal{CC}$-like sets have to be considered instead of $\mathcal{SC}$-like sets. ■

Lemma 5.45 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$ and $\alpha \in \mathit{Name}^* - \{\varepsilon\}$. Then for all $\theta \in \mathbb{R}^+_{>0}$:

$$prob(\mathcal{CC}_{\leq \theta}(P_1 + P_2, \alpha)) = \begin{cases} p_1 \cdot prob(\mathcal{CC}_{\leq \theta_1}(P_1, \alpha)) + p_2 \cdot prob(\mathcal{CC}_{\leq \theta_2}(P_2, \alpha)) \\ \qquad \text{if } r_1 > 0 \wedge r_2 > 0 \\ prob(\mathcal{CC}_{\leq \theta}(P_1, \alpha)) \\ \qquad \text{if } r_1 > 0 \wedge r_2 = 0 \\ prob(\mathcal{CC}_{\leq \theta}(P_2, \alpha)) \\ \qquad \text{if } r_1 = 0 \wedge r_2 > 0 \\ 0 \\ \qquad \text{if } r_1 = 0 \wedge r_2 = 0 \end{cases}$$

where:

$$r_1 = rate_{\mathrm{t}}(P_1, 0) \qquad r_2 = rate_{\mathrm{t}}(P_2, 0)$$

$$p_1 = \frac{r_1}{r_1+r_2} \qquad p_2 = \frac{r_2}{r_1+r_2}$$

$$\theta_1[i] = \begin{cases} \theta[i] + (\frac{1}{r_1} - \frac{1}{r_1+r_2}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases} \qquad \theta_2[i] = \begin{cases} \theta[i] + (\frac{1}{r_2} - \frac{1}{r_1+r_2}) & \text{if } i = 1 \\ \theta[i] & \text{if } i > 1 \end{cases}$$

Proof Similar to the proof of Lemma 4.52, with the difference that $\mathcal{CC}$-like sets have to be considered instead of $\mathcal{SC}$-like sets. ■

Theorem 5.46 Let $P_1, P_2 \in \mathcal{P}_{\mathrm{S}}$. Whenever $P_1 \sim_{\mathrm{MTr}} P_2$, then:

(1) $<a, \lambda>.P_1 \sim_{\mathrm{MTr}} <a, \lambda>.P_2$ for all $<a, \lambda> \in Act_{\mathrm{S}}$.
(2) $P_1 + P \sim_{\mathrm{MTr}} P_2 + P$ and $P + P_1 \sim_{\mathrm{MTr}} P + P_2$ for all $P \in \mathcal{P}_{\mathrm{S}}$.

Proof (1) In order to avoid trivial cases, consider $\alpha \equiv a \circ \alpha'$. From $P_1 \sim_{\mathrm{MTr}} P_2$ it follows that for all $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{CC}_{\leq \theta}(P_1, \alpha')) = prob(\mathcal{CC}_{\leq \theta}(P_2, \alpha'))$$

By virtue of Lemma 5.44, for each $k = 1, 2$ we have that for all $\theta \in \mathbb{R}^*_{>0}$:

$$prob(\mathcal{CC}_{\leq \theta}(P_k, \alpha')) = prob(\mathcal{CC}_{\leq \frac{1}{\lambda} \circ \theta}(<a, \lambda>.P_k, a \circ \alpha'))$$

hence for all $\theta' = \frac{1}{\mu} \circ \theta \in \mathbb{R}^+_{>0}$ such that $\frac{1}{\mu} \geq \frac{1}{\lambda}$:

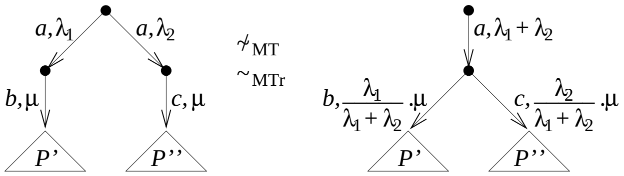
$$prob(\mathcal{CC}_{\leq \theta'}(<a, \lambda>.P_1, \alpha)) = prob(\mathcal{CC}_{\leq \theta'}(<a, \lambda>.P_2, \alpha))$$

Since $\alpha \not\equiv \varepsilon$, for $\theta' = \varepsilon$ and for all $\theta' \in \mathbb{R}^+_{>0}$ such that $\theta'[1] < \frac{1}{\lambda}$ we have:

$$prob(\mathcal{CC}_{\leq \theta'}(<a, \lambda>.P_1, \alpha)) = 0 = prob(\mathcal{CC}_{\leq \theta'}(<a, \lambda>.P_2, \alpha))$$

(2) Similar to the proof of Thm. 4.53(2), with the difference that $\mathcal{CC}$-like sets have to be considered instead of $\mathcal{SC}$-like sets, and that Lemma 5.45 and Cor. 5.6 have to be used. ■

As shown by Prop. 5.30 and Ex. 5.31, $\sim_{\rm MT}$ is strictly contained in $\sim_{\rm MTr}$, hence the axioms $\mathcal{A}_1^{\rm MT}$-$\mathcal{A}_4^{\rm MT}$ of Table 2 are still valid for $\sim_{\rm MTr}$ over $\mathcal{P}_{\rm S,nr}$, but not complete. In fact, the two process terms considered in Ex. 5.31, which are depicted below:



show that, when moving from $\sim_{\rm MT}$ to $\sim_{\rm MTr}$, the action prefix operator tends to become left-distributive with respect to the alternative composition operator. More precisely, choices can be deferred as long as they are related to branches starting with actions having the same name that are followed by terms having the same total exit rate. Note that the names and the total rates of the initial actions of such derivative terms can be different in the various branches.

$$(\mathcal{A}_1^{\rm MTr}) \quad P_1 + P_2 = P_2 + P_1$$

$$(\mathcal{A}_2^{\rm MTr}) \quad (P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$$

$$(\mathcal{A}_3^{\rm MTr}) \quad P + \underline{0} = P$$

$$(\mathcal{A}_4^{\rm MTr}) \quad \sum_{i \in I} <a, \lambda_i>. \sum_{j \in J_i} <b_{i,j}, \mu_{i,j}>.P_{i,j} = \\ <a, \textstyle\sum_{k \in I} \lambda_k>. \displaystyle\sum_{i \in I} \sum_{j \in J_i} <b_{i,j}, \frac{\lambda_i}{\sum_{k \in I} \lambda_k} \cdot \mu_{i,j}>.P_{i,j}$$

if for all $i_1, i_2 \in I$:

$$\sum_{j \in J_{i_1}} \mu_{i_1,j} = \sum_{j \in J_{i_2}} \mu_{i_2,j} \equiv \mu$$

Table 3
Axiomatization of $\sim_{\rm MTr}$ over $\mathcal{P}_{\rm S,nr}$

Here we prove that the two terms above constitute the simplest instance of a more liberal axiom schema $\mathcal{A}_4^{\rm MTr}$ that we have to substitute for the more restrictive axiom schema $\mathcal{A}_4^{\rm MT}$ in order to obtain a sound and complete axiomatization of $\sim_{\rm MTr}$ over $\mathcal{P}_{\rm S,nr}$. Such an axiomatization is given by the set $\mathcal{A}^{\rm MTr}$ of axioms shown in Table 3, where I and J_i are finite index sets with $|I| \geq 2$ (if $J_i = \emptyset$, the related summations are taken to be $\underline{0}$).

Theorem 5.47 The deduction system $DED(\mathcal{A}^{\rm MTr})$ is sound for $\sim_{\rm MTr}$ over $\mathcal{P}_{\rm S,nr}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\rm S,nr}$:

$$\mathcal{A}^{\rm MTr} \vdash P_1 = P_2 \implies P_1 \sim_{\rm MTr} P_2$$

Proof Similar to the proof of Thm. 4.54, with the difference that Thm. 5.46 and Prop. 5.30 have to be used, and that for $\mathcal{A}_4^{\mathrm{MTr}}$ it suffices to observe what follows in the case that $J_i \neq \emptyset$ for all $i \in I$:

- The union of the sets of the names of the actions initially executable by the $|I|$ a-derivative terms on the left, i.e. $\{b_{i,j} \mid i \in I \wedge j \in J_i\}$, coincides with the set of the names of the actions initially executable by the only a-derivative term on the right.
- Each of the a-derivative terms has the same total exit rate μ.
- For all traces starting with a followed by $b_{i,j}$, the a-$b_{i,j}$-derivative term $P_{i,j}$ is reached with the same probability $(\lambda_i / \sum_{k \in I} \lambda_k) \cdot (\mu_{i,j}/\mu)$ both on the left and on the right. ■

Definition 5.48 Let $P \in \mathcal{P}_{\mathrm{S,nr}}$. We say that P is in trace-minimal sum normal form (trmsnf) iff $P \equiv \underline{0}$ or $P \equiv \sum_{i \in I} {<}a_i, \lambda_i{>}.P_i$ with I finite and nonempty, P initially minimal with respect to $\mathcal{A}_4^{\mathrm{MTr}}$, and P_i in trmsnf for all $i \in I$. ■

Lemma 5.49 For all $P \in \mathcal{P}_{\mathrm{S,nr}}$ there exists $Q \in \mathcal{P}_{\mathrm{S,nr,trmsnf}}$ such that $\mathcal{A}^{\mathrm{MTr}} \vdash P = Q$.

Proof Similar to the proof of Lemma 4.56. ■

Theorem 5.50 The deduction system $DED(\mathcal{A}^{\mathrm{MTr}})$ is complete for $\sim_{\mathrm{MTr}}$ over $\mathcal{P}_{\mathrm{S,nr}}$, i.e. for all $P_1, P_2 \in \mathcal{P}_{\mathrm{S,nr}}$:

$$P_1 \sim_{\mathrm{MTr}} P_2 \implies \mathcal{A}^{\mathrm{MTr}} \vdash P_1 = P_2$$

Proof Similar to the proof of Thm. 4.57, with the following differences.
In the case that P_1 and P_2 are both in trmsnf, with $P_1 \equiv \sum_{i \in I_1} {<}a_i, \lambda_i{>}.P_{1,i}$ and $P_2 \equiv \sum_{j \in I_2} {<}b_j, \mu_j{>}.P_{2,j}$, recalled that by virtue of Cor. 5.5 from $P_1 \sim_{\mathrm{MTr}} P_2$ it follows that:

$$\{a_i \mid i \in I_1\} = \{b_j \mid j \in I_2\} \equiv \{c_1, c_2, \ldots, c_n\}$$

with:

$$rate(P_1, c_k, 0, \mathcal{P}_{\mathrm{C}}) = rate(P_2, c_k, 0, \mathcal{P}_{\mathrm{C}})$$

for each $k = 1, \ldots, n$, we have that for a generic c_k and the two related sets of summands:

$$S_{k,1} = \{{<}a_i, \lambda_i{>}.P_{1,i} \mid i \in I_1 \wedge a_i = c_k\}$$

$$S_{k,2} = \{{<}b_j, \mu_j{>}.P_{2,j} \mid j \in I_2 \wedge b_j = c_k\}$$

the following two properties are satisfied:

(1) $\sum_{P \in S_{k,1}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}}) = \sum_{P \in S_{k,2}} rate(P, c_k, 0, \mathcal{P}_{\mathrm{C}})$.

(2) The derivative terms $P_{1,i}$ (resp. $P_{2,j}$) occurring in $S_{k,1}$ (resp. $S_{k,2}$) are all inequivalent with respect to $\sim_{\rm MTr}$ due to the initial minimality of P_1 (resp. P_2) with respect to $\mathcal{A}_4^{\rm MTr}$. In fact, due to such an initial minimality, taken two derivative terms in the same summand set, it must be the case that their total exit rates are different, thus violating the necessary condition for $\sim_{\rm MTr}$ stated by Cor. 5.6.

In the proof by induction on $|S_{k,1}|$ that for each summand $<a_i, \lambda_i>.P_{1,i} \in S_{k,1}$ there exists exactly one summand $<b_j, \mu_j>.P_{2,j} \in S_{k,2}$ such that $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\rm MTr} P_{2,j}$, part of the demonstration has to be modified as follows:

- If $|S_{k,1}| = 1$, then $S_{k,1}$ contains a single summand, say $<a_i, \lambda_i>.P_{1,i}$. Then $S_{k,2}$ must contain a single summand as well, say $<b_j, \mu_j>.P_{2,j}$, with $\lambda_i = \mu_j \wedge P_{1,i} \sim_{\rm MTr} P_{2,j}$ (as $P_1 \sim_{\rm MTr} P_2$, hence P_1 and P_2 cannot be distinguished by traces starting with a c_k-action). The reason why $S_{k,2}$ cannot contain several summands – each starting with a c_k-action – is that this would contradict $P_1 \sim_{\rm MTr} P_2$ (in the case in which the inequivalent derivatives of the summands violated the necessary condition for $\sim_{\rm MTr}$ stated by Cor. 5.6) or the initial minimality of P_2 with respect to $\mathcal{A}_4^{\rm MTr}$ (in the case in which the inequivalent derivatives satisfied that necessary condition).
- Suppose that $|S_{k,1}| = m > 1$. Let $S_{k,1}^d$ be the set of the summands of $S_{k,1}$ whose derivative terms have – among all the derivative terms occurring in $S_{k,1}$ – the minimum average sojourn time $1/\Delta$ and the same total exit rate δ with respect to an action name d. By virtue of property (2), d can be chosen in such a way that $S_{k,1}^d \neq S_{k,1}$. Then the derivative term of each summand of $S_{k,1}^d$ executes with probability δ/Δ the trace d within the minimum average time $1/\Delta$, hence P_1 executes with probability $(\sum_{P \in S_{k,1}^d} rate(P, c_k, 0, \mathcal{P}_{\rm C}) / rate_{\rm t}(P_1, 0)) \cdot (\delta/\Delta)$ the trace $c_k \circ d$ within the minimum average time sequence $1/rate_{\rm t}(P_1, 0) \circ 1/\Delta$. Since $P_1 \sim_{\rm MTr} P_2$, also P_2 must execute the same trace in the same way as P_1, hence there must exist a subset $S_{k,2}^d$ of $S_{k,2}$ whose derivative terms all have the minimum average sojourn time $1/\Delta$ and the same total exit rate δ with respect to d, with $S_{k,2}^d \neq S_{k,2}$ and $\sum_{P \in S_{k,1}^d} rate(P, c_k, 0, \mathcal{P}_{\rm C}) = \sum_{P \in S_{k,2}^d} rate(P, c_k, 0, \mathcal{P}_{\rm C})$.

In the case that at least one between P_1 and P_2 is not in trmsnf, Lemma 5.49 and Thm. 5.47 have to be used. ■

5.6 Exact Aggregation Property

By looking at the structure and at the rate constraints of the axiom schemata $\mathcal{A}_4^{\rm MT}$ and $\mathcal{A}_4^{\rm MTr}$, it is straightforward to conclude that both axiom schemata result in the same CTMC-level aggregation, which is the one depicted at the beginning of Sect. 4.7.

Theorem 5.51 $\sim_{\text{MTr}}$ induces the same CTMC-level aggregation as $\sim_{\text{MT}}$. ■

Corollary 5.52 The CTMC-level aggregation induced by $\sim_{\text{MTr}}$ is exact. ■

6 Markovian Testing and Trace Equivalences for CMPC

In this section we consider $\sim_{\text{MT}}$ and $\sim_{\text{MTr}}$ over CMPC. Since passive actions now come into play through the syntax of the process terms, the definitions of $\sim_{\text{MT}}$ and $\sim_{\text{MTr}}$ may become more complicated. The reason is that the presence of passive actions that do not have to synchronize with exponentially timed actions may hamper the calculation of the execution probability and of the duration of test-driven and non-test-driven computations.

We shall therefore restrict ourselves to the performance closed process terms of CMPC. Since these terms are guaranteed not to execute any passive action that does not synchronize with an exponentially timed action, the definitions of $\sim_{\text{MT}}$ and $\sim_{\text{MTr}}$ will not need any change and all the results of Sect. 4 and Sect. 5 will carry over. In this performance closed setting we shall show that $\sim_{\text{MT}}$ is a congruence with respect to parallel composition whereas $\sim_{\text{MTr}}$ is not, from which it can be concluded that only Markovian testing equivalence may constitute a useful alternative to Markovian bisimilarity.

6.1 Full Congruence Property of $\sim_{\text{MT}}$

We now prove that $\sim_{\text{MT}}$ is a congruence over $\mathcal{P}_{\text{C,pc}}$ with respect to parallel composition as long as the context process put in parallel can only perform synchronizing passive actions. The proof will be accomplished by exploiting the extended-trace-based alternative characterization of $\sim_{\text{MT}}$ stemming from the full abstraction result of Sect. 4.3.4.

Theorem 6.1 Let $P_1, P_2 \in \mathcal{P}_{\text{C,pc}}$, $P \in \mathcal{P}_{\text{C}}$ such that it contains only passive actions, and $S \subseteq \mathit{Name}$ such that $P_1 \,\|_S\, P, P_2 \,\|_S\, P, P \,\|_S\, P_1, P \,\|_S\, P_2 \in \mathcal{P}_{\text{C,pc}}$. Whenever $P_1 \sim_{\text{MT}} P_2$, then:

$$P_1 \,\|_S\, P \sim_{\text{MT}} P_2 \,\|_S\, P$$

$$P \,\|_S\, P_1 \sim_{\text{MT}} P \,\|_S\, P_2$$

Proof We start by observing that any computation $c \in \mathcal{C}_{\text{f}}(P_k \,\|_S\, P)$, with $k \in \{1, 2\}$, is the parallel composition with respect to S of a computation $c_{P_k} \in \mathcal{C}_{\text{f}}(P_k)$ and a computation $c_P \in \mathcal{C}_{\text{f}}(P)$. Since P contains only passive actions and $P_k \,\|_S\, P$ is performance closed, we have:

$$c = c_{P_k} \,\|_S\, c_P = \begin{cases} P_k \,\|_S\, P \xrightarrow{a,\lambda} c_{P'_k} \,\|_S\, c_P \\ \qquad \text{if } c_{P_k} \equiv P_k \xrightarrow{a,\lambda} c_{P'_k} \wedge a \notin S \\ P_k \,\|_S\, P \xrightarrow{a,\lambda \cdot \frac{w}{weight(P,a)}} c_{P'_k} \,\|_S\, c_{P'} \\ \qquad \text{if } c_{P_k} \equiv P_k \xrightarrow{a,\lambda} c_{P'_k} \wedge c_P \equiv P \xrightarrow{a,*w} c_{P'} \wedge a \in S \\ P_k \,\|_S\, P \\ \qquad \text{otherwise} \end{cases}$$

For all extended traces $\sigma \in \mathcal{ET}$ and for all computations $c = c_{P_k} \,\|_S\, c_P \in \mathcal{CC}(P_k \,\|_S\, P, \sigma)$ we have:

$$prob^{\sigma}(c) = \begin{cases} 1 \\ \qquad \text{if } length(c) = 0 \\ \frac{\lambda}{\sum\limits_{b \in \mathcal{E}} \{\!| rate(P_k, b, 0, \mathcal{P}_{\mathrm{C}}) \mid b \notin S \vee weight(P,b) > 0 |\!\}} \cdot prob^{\sigma'}(c') \\ \qquad \text{if } c \equiv P_k \,\|_S\, P \xrightarrow{a,\lambda} c' \text{ with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

and:

$$time_{\mathrm{a}}^{\sigma}(c) = \begin{cases} \varepsilon \\ \qquad \text{if } length(c) = 0 \\ \frac{1}{\sum\limits_{b \in \mathcal{E}} \{\!| rate(P_k, b, 0, \mathcal{P}_{\mathrm{C}}) \mid b \notin S \vee weight(P,b) > 0 |\!\}} \circ time_{\mathrm{a}}^{\sigma'}(c') \\ \qquad \text{if } c \equiv P_k \,\|_S\, P \xrightarrow{a,\lambda} c' \text{ with } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \end{cases}$$

Let us define the environment-related restriction of the above mentioned σ to c_P and S through the following $\mathcal{ET}$-valued function:

$$restr(\sigma, c_P, S) = \begin{cases} (a, \{b \in \mathcal{E} \mid b \notin S \vee weight(P,b) > 0\}) \circ restr(\sigma', c_P, S) \\ \qquad \text{if } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \wedge a \notin S \\ (a, \{b \in \mathcal{E} \mid b \notin S \vee weight(P,b) > 0\}) \circ restr(\sigma', c_{P'}, S) \\ \qquad \text{if } \sigma \equiv (a, \mathcal{E}) \circ \sigma' \wedge a \in S \wedge c_P \equiv P \xrightarrow{a,*w} c_{P'} \end{cases}$$

Then for all $c = c_{P_k} \,\|_S\, c_P \in \mathcal{CC}(P_k \,\|_S\, P, \sigma)$:

$$\begin{aligned} prob^{\sigma}(c) &= prob^{restr(\sigma, c_P, S)}(c_{P_k}) \\ time_{\mathrm{a}}^{\sigma}(c) &= time_{\mathrm{a}}^{restr(\sigma, c_P, S)}(c_{P_k}) \end{aligned}$$

As a consequence, for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^{*}_{>0}$:

$$prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_k \parallel_S P, \sigma)) = \sum_{c = c_{P_k} \parallel_S c_P \in \mathcal{CC}^{\sigma}_{\leq\theta}(P_k \parallel_S P, \sigma)} prob^{\sigma}(c)$$

$$= \sum_{c_P \in \mathcal{C}_{\mathrm{f}}(P)} \sum_{c_{P_k} \in \mathcal{CC}^{restr(\sigma, c_P, S)}_{\leq\theta}(P_k, restr(\sigma, c_P, S))} prob^{restr(\sigma, c_P, S)}(c_{P_k})$$

$$= \sum_{c_P \in \mathcal{C}_{\mathrm{f}}(P)} prob^{restr(\sigma, c_P, S)}(\mathcal{CC}^{restr(\sigma, c_P, S)}_{\leq\theta}(P_k, restr(\sigma, c_P, S)))$$

From $P_1 \sim_{\mathrm{MT}} P_2$ and Thm. 4.35 we derive that for all $\sigma \in \mathcal{ET}$, $c_P \in \mathcal{C}_{\mathrm{f}}(P)$, and $\theta \in \mathbb{R}^*_{>0}$:

$$prob^{restr(\sigma, c_P, S)}(\mathcal{CC}^{restr(\sigma, c_P, S)}_{\leq\theta}(P_1, restr(\sigma, c_P, S))) = prob^{restr(\sigma, c_P, S)}(\mathcal{CC}^{restr(\sigma, c_P, S)}_{\leq\theta}(P_2, restr(\sigma, c_P, S)))$$

hence for all $\sigma \in \mathcal{ET}$ and $\theta \in \mathbb{R}^*_{>0}$:

$$prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_1 \parallel_S P, \sigma)) = prob^{\sigma}(\mathcal{CC}^{\sigma}_{\leq\theta}(P_2 \parallel_S P, \sigma))$$

The result then follows by virtue of Thm. 4.35. ■

6.2 $\sim_{\mathrm{MTr}}$ *Is Not a Full Congruence*

Unlike $\sim_{\mathrm{MT}}$, it turns out that $\sim_{\mathrm{MTr}}$ is not a congruence with respect to parallel composition. This should not come as a surprise, as in [18] it has been shown that probabilistic trace equivalence is not a congruence with respect to parallel composition.

Consider the two Markovian trace equivalent process terms of Ex. 5.31:

$$P_7 \equiv <a, \lambda_1>.<b, \mu>.P' + <a, \lambda_2>.<c, \mu>.P''$$

$$P_8 \equiv <a, \lambda_1 + \lambda_2>.(<b, \tfrac{\lambda_1}{\lambda_1+\lambda_2} \cdot \mu>.P' + <c, \tfrac{\lambda_2}{\lambda_1+\lambda_2} \cdot \mu>.P'')$$

where $b \neq c$. If we place each of them in the following context:

$$\_ \parallel_{\{a,b,c\}} <a, *_1>.<b, *_1>.\underline{0}$$

we obtain two performance closed process terms – which we call Q_7 and Q_8 – that are no longer Markovian trace equivalent.

The following trace:

$$\alpha \equiv a \circ b$$

can in fact distinguish between Q_7 and Q_8. The reason is that the only computation of Q_7 compatible with α is formed by a transition labeled with $<a, \lambda_1>$ followed by a transition labeled with $<b, \mu>$, which has execution probability $\frac{\lambda_1}{\lambda_1+\lambda_2}$ and average duration $\frac{1}{\lambda_1+\lambda_2} \circ \frac{1}{\mu}$. By contrast, the only computation of Q_8 compatible with α is formed by a transition labeled with $<a, \lambda_1+\lambda_2>$ followed by a transition labeled with $<b, \frac{\lambda_1}{\lambda_1+\lambda_2} \cdot \mu>$, which has execution probability 1 and average duration $\frac{1}{\lambda_1+\lambda_2} \circ \frac{\lambda_1+\lambda_2}{\lambda_1 \cdot \mu}$.

7 Conclusion

In this paper we have shown that Markovian testing and trace equivalences induce at the CTMC level the same aggregation, which is strictly coarser than ordinary lumping and exact. This ensures that, whenever two process terms are Markovian testing or trace equivalent, they possess the same performance characteristics. Another consequence is that Markovian testing and trace equivalences improve on Markovian bisimilarity in terms of state space reduction, while preserving the exact aggregation property.

From a concurrency theory perspective, the exact aggregation result of this paper, together with the exact aggregation result of [16], establishes that all the three major approaches to the definition of behavioral equivalences – bisimulation, testing, trace – are sound from the stochastic viewpoint, hence useful for performance evaluation purposes.

Viewed from a different angle, the main result of this paper is – to the best of our knowledge – the discovery of a new exact aggregation in the Markov chain theory, which is strictly coarser than ordinary lumping and entirely characterized in a process algebraic setting like ordinary lumping.

The strategy adopted to prove the main result has led to demonstrate two further results related to Markovian testing and trace equivalences over a sequential Markovian process calculus. The first result is that the two equivalences are congruences with respect to action prefix and alternative composition. The second result is that the two equivalences have sound and complete axiomatizations.

In addition to that, we have found several alternative characterizations of Markovian testing and trace equivalences – which justify the use of average durations instead of duration distributions and include a full abstraction result together with the identification of a set of canonical tests – as well as a number of connections with other behavioral equivalences – which in particular provides information about the Markovian linear-time/branching-time spectrum. Finally, we have proved that Markovian testing equivalence is a congruence with respect to (a restricted version of) parallel composition, whereas Markovian trace equivalence is not, thus discovering that only Markovian testing equivalence may constitute a useful alternative to Markovian bisimilarity.

After establishing the fundamental property of exact aggregation, it becomes meaningful to investigate further properties – besides the already addressed congruence and axiomatization – of Markovian testing and trace equivalences.

On the theoretical side, first of all we would like to permit invisible actions within process terms and study to what extent the two equivalences would be able to abstract from such actions while preserving compositionality. Note that the probabilistic nature of our Markovian framework and the presence of time upper bounds would avoid problems with divergence in the testing case. Second, we would like to see what happens when admitting invisible actions within tests. Our conjecture is that this should make the time upper bounds useless from the point of view of the distinguishing power. Third, we would like to extend the definitions of the two equivalences to process terms that are not performance closed, which should by the way be useful to extend the congruence result with respect to parallel composition in the testing case. This extension may be accomplished by replacing the passive rate of every non-synchronizing passive action with a symbolic exponential rate that somehow encodes the weight of the passive action. Fourth, we would like to derive modal/temporal logic characterizations for the two equivalences. Some work in this direction can be found in [5].

On the verification side, a crucial issue – especially for Markovian testing equivalence – in order to set up a useful alternative to Markovian bisimilarity is to find out efficient algorithms. We believe that a good starting point for Markovian testing equivalence may be the algorithm for classical testing equivalence proposed in [11]. This requires a more denotational characterization of Markovian testing equivalence, which may be inspired by the probabilistic variant of the acceptance tree model proposed in [21]. The issue of checking two process terms for Markovian trace equivalence has already been addressed in [23], where a polynomial-time algorithm has been devised. The derivation of such algorithms should also lead to an effective way to minimize CTMCs according to the newly discovered exact aggregation.

Furthermore, we would like to assess whether Markovian testing equivalence can be used for quantitative purposes as well. So far, it supports a merely qualitative analysis, in the sense that – based on its generic notion of efficiency related to the probability with which the tests are passed within a certain average amount of time – it only allows one to establish whether two process terms possess the same performance characteristics or not. What we envision is the possibility of identifying classes of tests that are related to specific performability measures, which may thus be used to evaluate process terms with respect to certain indices of interest.

As a final remark, in the light of the exact aggregation property proved in this paper for Markovian testing and trace equivalences, which in a sense extends ordinary lumping, it becomes interesting to understand whether the CTMC-level aggregation induced by such equivalences is the coarsest exact one that can be obtained, or whether it can be further extended.

Acknowledgements

We wish to thank Rance Cleaveland for the collaboration in the years 1997-2000 on Markovian testing equivalence. We also wish to mention that the exactness of the CTMC-level aggregation induced by Markovian testing equivalence was conjectured in 2000, at the end of a discussion with Mario Bravetti. Finally, we would like to thank the anonymous referees for their constructive comments and their helpful suggestions.

References

[1] J.C.M. Baeten and W.P. Weijland, *"Process Algebra"*, Cambridge University Press, 1990.

[2] C. Baier, J.-P. Katoen, H. Hermanns, and V. Wolf, *"Comparative Branching-Time Semantics for Markov Chains"*, in Information and Computation 200:149-214, 2005.

[3] J.A. Bergstra, A. Ponse, and S.A. Smolka (eds.), *"Handbook of Process Algebra"*, Elsevier, 2001.

[4] M. Bernardo, *"Markovian Testing and Trace Equivalences Exactly Lump More Than Markovian Bisimilarity"*, in Proc. of the *Int. Workshop on Algebraic Process Calculi: The First Twenty Five Years and Beyond (APC 25)*, ENTCS, Bertinoro (Italy), 2005.

[5] M. Bernardo and S. Botta, *"Modal Logic Characterization of Markovian Testing and Trace Equivalences"*, in Proc. of the *1st Int. Workshop on Logic, Models and Computer Science (LMCS 2006)*, ENTCS, Camerino (Italy), 2006.

[6] M. Bernardo and M. Bravetti, *"Performance Measure Sensitive Congruences for Markovian Process Algebras"*, in Theoretical Computer Science 290:117-160, 2003.

[7] M. Bernardo and R. Cleaveland, *"A Theory of Testing for Markovian Processes"*, in Proc. of the *11th Int. Conf. on Concurrency Theory (CONCUR 2000)*, LNCS 1877:305-319, State College (PA), 2000.

[8] P. Buchholz, *"Exact and Ordinary Lumpability in Finite Markov Chains"*, in Journal of Applied Probability 31:59-75, 1994.

[9] I. Christoff, *"Testing Equivalences and Fully Abstract Models for Probabilistic Processes"*, in Proc. of the *1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, LNCS 458:126-140, Amsterdam (The Netherlands), 1990.

[10] R. Cleaveland, Z. Dayar, S.A. Smolka, and S. Yuen, *"Testing Preorders for Probabilistic Processes"*, in Information and Computation 154:93-148, 1999.

[11] R. Cleaveland and M. Hennessy, *"Testing Equivalence as a Bisimulation Equivalence"*, in Formal Aspects of Computing 5:1-20, 1993.

[12] R. De Nicola and M. Hennessy, *"Testing Equivalences for Processes"*, in Theoretical Computer Science 34:83-133, 1983.

[13] R.J. van Glabbeek, S.A. Smolka, and B. Steffen, *"Reactive, Generative and Stratified Models of Probabilistic Processes"*, in Information and Computation 121:59-80, 1995.

[14] H. Hermanns, *"Interactive Markov Chains"*, LNCS 2428, 2002.

[15] H. Hermanns and M. Rettelbach, *"Syntax, Semantics, Equivalences, and Axioms for MTIPP"*, in Proc. of the *2nd Int. Workshop on Process Algebra and Performance Modelling (PAPM 1994)*, Technical Report 27-4, pp. 71-87, Erlangen (Germany), 1994.

[16] J. Hillston, *"A Compositional Approach to Performance Modelling"*, Cambridge University Press, 1996.

[17] C.A.R. Hoare, *"Communicating Sequential Processes"*, Prentice Hall, 1985.

[18] C.-C. Jou and S.A. Smolka, *"Equivalences, Congruences, and Complete Axiomatizations for Probabilistic Processes"*, in Proc. of the *1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, LNCS 458:367-383, Amsterdam (The Netherlands), 1990.

[19] K.G. Larsen and A. Skou, *"Bisimulation through Probabilistic Testing"*, in Information and Computation 94:1-28, 1991.

[20] R. Milner, *"Communication and Concurrency"*, Prentice Hall, 1989.

[21] M. Nuñez, *"Algebraic Theory of Probabilistic Processes"*, in Journal of Logic and Algebraic Programming 56:117-177, 2003.

[22] E.W. Stark, R. Cleaveland, and S.A. Smolka, *"A Process-Algebraic Language for Probabilistic I/O Automata"*, in Proc. of the *14th Int. Conf. on Concurrency Theory (CONCUR 2003)*, LNCS 2761:189-203, Marseille (France), 2003.

[23] V. Wolf, C. Baier, and M. Majster-Cederbaum, *"Trace Machines for Observing Continuous-Time Markov Chains"*, in Proc. of the *3rd Int. Workshop on Quantitative Aspects of Programming Languages (QAPL 2005)*, ENTCS 153(2):259-277, Edinburgh (UK), 2005.