Simulation for Continuous-Time Markov Chains

Christel Baier[a], Joost-Pieter Katoen[b],
Holger Hermanns[b] and Boudewijn Haverkort[c]

[a] *Institut für Informatik I, University of Bonn*
Römerstraße 164, D-53117 Bonn, Germany

[c] *Faculty of Computer Science, University of Twente*
P.O. Box 217, 7500 AE Enschede, The Netherlands

[b] *Dept. of Computer Science, RWTH Aachen*
Ahornstraße 55, D-52056 Aachen, Germany

Abstract. This paper presents a simulation preorder for continuous-time Markov chains (CTMCs). The simulation preorder is a conservative extension of a weak variant of probabilistic simulation on fully probabilistic systems, i.e., discrete-time Markov chains. The main result of the paper is that the simulation preorder preserves safety and liveness properties expressed in continuous stochastic logic (CSL), a stochastic branching-time temporal logic interpreted over CTMCs.

1 Introduction

To compare the stepwise behaviour of states in transition systems, simulation ($\sqsubseteq$) and bisimulation relations ($\sim$) have been widely considered [31, 25]. Bisimulation relations are equivalences such that two bisimilar states exhibit identical stepwise behaviour. On the contrary, simulation relations are preorders on the state space such that if $s \sqsubseteq s'$ ("s' simulates s") state s' can mimic all stepwise behaviour of s; the converse, i.e., $s' \sqsubseteq s$ is not guaranteed, so state s' may perform steps that cannot be matched by s. Thus, if $s \sqsubseteq s'$ then every successor of s has a corresponding, i.e., related successor of s', but the reverse does not necessarily hold. Simulation can be lifted to entire transition systems by comparing (according to $\sqsubseteq$) their initial states. Simulation relations are often used for verification purposes to show that one system correctly implements another, more abstract system. One of the interesting aspects of simulation relations is that they allow a verification by "local" reasoning.

In the setting of model checking, (bi)simulation relations can be used to combat the well-known state-space explosion problem [15]. Here, bisimulation relations possess the so-called *strong preservation* property, whereas simulation possesses *weak preservation*. Strong preservation means: if $s \sim s'$, then for all formulas Φ it follows $s \models \Phi$ iff $s' \models \Phi$. This property holds, for instance, for CTL (and CTL*) and strong bisimulation [12]. The use of simulation relies on the preservation of certain classes of formulas, not of all formulas (such as for $\sim$). For instance, if $s \sqsubseteq s'$ then for all safety formulas Φ it follows that $s' \models \Phi$

implies $s \models \Phi$.[1] Note that the converse, $s' \not\models \Phi$, cannot be used to deduce that Φ does not hold in the simulated state s; hence, the name *weak* preservation. As simulation equivalence – defined as mutual simulation of states – is coarser than bisimulation equivalence it yields a "better abstraction", i.e., a smaller quotient. Simulation relations are the basis for abstraction techniques where the rough idea is to replace the large system to be verified by a small abstract model and to model check the abstract system.

This paper studies a simulation preorder for continuous-time Markov chains (CTMCs) [29, 36] and investigates the preservation of properties expressed in continuous stochastic logic (CSL) [3, 8]. CTMCs are an important class of stochastic processes that are widely used in practice to determine system performance and dependability characteristics. CSL is a continuous probabilistic variant of CTL and includes means to express both transient and steady-state performance measures. For instance, it allows one to stipulate that the probability of reaching a certain set of goal-states within a specified real-valued time bound, provided that all paths to these states obey certain properties, is at least/at most some probability value. Model-checking algorithms for CSL have been presented in [8, 6], and prototypical software implementations are available: one based on sparse matrices [23] and a symbolic one based on multi-terminal BDDs [28]. Baier *et al.* [6] prove that lumping equivalence [13] – a continuous time variant of probabilistic bisimulation – preserves CSL; Desharnais and Panangaden [18] have recently shown the converse, namely that the equivalence induced by CSL implies lumping equivalence.

This paper proposes a novel simulation preorder ($\sqsubseteq_m$) for CTMCs. This notion extends probabilistic simulation ($\sqsubseteq_p$) on discrete-time Markov chains (DTMCs), as originally defined by Jonsson and Larsen [26]. The main result of the paper is that $\sqsubseteq_m$ weakly preserves CSL safety and liveness properties. This means that for $s \sqsubseteq_m s'$ we have that $s' \models \Phi_{safe}$ implies $s \models \Phi_{safe}$ for any CSL safety-formula Φ_{safe}, and that $s \models \Phi_{live}$ implies $s' \models \Phi_{live}$ for any CSL liveness-formula Φ_{live}. As a consequence, the validity of safety formulas and the refutation of liveness formulas carries over from the abstract state s' (wrt. $\sqsubseteq_m$) to the concrete state s. This result can be used to verify CSL-formulas for a CTMC by verifying the same formulas on a smaller or simpler CTMC which is an abstraction of it.

Organisation of the paper. Section 2 introduces CTMCs, presents the simulation preorder for CTMCs and some of its elementary properties. Section 3 recalls CSL and introduces its safe and live fragments. Section 4 discusses weak preservation of CSL-formulas. Section 5 defines simulation equivalence and compares this to other equivalence notions. Section 6 discusses related work. Section 7 concludes the paper. Proofs of the main results are provided in [9].

[1] Safety formulas are here to be understood as arbitrary formulas in $\forall$CTL*, the restriction of CTL* to universal path-quantifiers [14].

2 Simulation for CTMCs

Fully probabilistic systems. Let AP be a fixed, finite set of atomic propositions. A (labelled) fully probabilistic system (FPS) $\mathcal{D}$ is a tuple $(S, \mathbf{P}, L)$ where S is a countable set of *states*, $\mathbf{P} : S \times S \to [0,1]$ is a *probability matrix* satisfying $\sum_{s' \in S} \mathbf{P}(s,s') \in [0,1]$ for all $s \in S$, and $L : S \to 2^{AP}$ is a *labelling* function which assigns to each state $s \in S$ the set $L(s)$ of atomic propositions that are valid in s. If $\sum_{s' \in S} \mathbf{P}(s,s') = 1$ for all $s \in S$, then $\mathbf{P}(s,\cdot)$ (and $\mathcal{D}$) is called stochastic, otherwise it is called sub-stochastic. A (labelled) DTMC is an FPS with $\sum_{s' \in S} \mathbf{P}(s,s') \in \{0,1\}$ for all $s \in S$.

Continuous-time Markov chains. A (labelled) CTMC $\mathcal{M}$ is a tuple $(S, \mathbf{R}, L)$ where S and L are as before, and $\mathbf{R} : S \times S \to \mathbb{R}_{\geqslant 0}$ is the *rate matrix*. (We adopt the same conventions as in [6,8], i.e., we do allow self-loops.) The exit rate $E(s) = \sum_{s' \in S} \mathbf{R}(s,s')$ denotes that the probability of taking a transition from s within t time units equals $1 - e^{-E(s) \cdot t}$. If $\mathbf{R}(s,s') > 0$ for more than one state s', a *race* between the outgoing transitions from s exists. That is, the probability $\mathbf{P}(s,s')$ of moving from s to s' in a single step equals the probability that the delay of going from s to s' "finishes before" the delays of any other outgoing transition from s; i.e., $\mathbf{P}(s,s') = \mathbf{R}(s,s')/E(s)$ if $E(s) > 0$ and 0 otherwise.

Definition 1. *For CTMC $\mathcal{M} = (S, \mathbf{R}, L)$, the* embedded *discrete-time Markov chain is given by* $\mathrm{emb}(\mathcal{M}) = (S, \mathbf{P}, L)$*, where* $\mathbf{P}(s,s') = \mathbf{R}(s,s')/E(s)$ *if* $E(s) > 0$*, and* $\mathbf{P}(s,s) = 1$ *and* $\mathbf{P}(s,s') = 0$ *for* $s \neq s'$ *if* $E(s) = 0$.

Note that, by definition, the embedded DTMC $emb(\mathcal{M})$ of any CTMC $\mathcal{M}$ is stochastic, i.e., $\sum_{s'} P(s,s') = 1$ for any state s.

Definition 2. *For CTMC $\mathcal{M} = (S, \mathbf{R}, L)$ the* uniformised *CTMC is given by* $unif(\mathcal{M}) = (S, \overline{\mathbf{R}}, L)$ *where* $\overline{\mathbf{R}}(s,s') = \mathbf{R}(s,s')$ *for* $s \neq s'$ *and* $\overline{\mathbf{R}}(s,s) = \mathbf{R}(s,s) + E - E(s)$ *where constant* $E \geqslant \max_{s \in S} E(s)$.

E is called the *uniformisation rate* of $\mathcal{M}$, and is determined by the state with the shortest mean residence time, since $E \geqslant max_{s \in S} E(s)$. All rates of self-loops in the CTMC $\mathcal{M}$ are "normalised" with respect to E, and hence the mean residence time is uniformly set to $1/E$ in $unif(\mathcal{M})$. In the literature [21, 24], uniformisation is often defined by transforming CTMC $\mathcal{M}$ into the DTMC $emb(unif(\mathcal{M}))$. For technical convenience, we here define uniformisation as a transformation from CTMCs to CTMCs basically by adding self-loops to slower states (as e.g. in [33]).

Simulation for fully probabilistic systems. For labelled transition systems, state s' simulates s if for each successor t of s there is a successor t' of s' that simulates t. Simulation of two states is thus defined in terms of simulation of their successor states. In the probabilistic setting, the target of a transition is in fact a probability distribution, and thus, the simulation relation $\sqsubseteq$ needs to be lifted from states to distributions. This can be done using *weight functions* [26]. For countable set X, let $Dist(X)$ denote the collection of all, possibly sub-stochastic, distributions on X.

Definition 3. *Let $\mu \in Dist(X)$ and $\mu' \in Dist(Y)$ and $\sqsubseteq\, \subseteq X \times Y$. Then $\mu \preceq \mu'$ iff there exists a weight function $\Delta : X \times Y \to [0,1]$ for $\sqsubseteq$ such that:*

1. *$\Delta(x,y) > 0$ implies $x \sqsubseteq y$*
2. *$\mu(x) = K_1 \cdot \sum_{y \in Y} \Delta(x,y)$ for any $x \in X$*
3. *$\mu'(y) = K_2 \cdot \sum_{x \in X} \Delta(x,y)$ for any $y \in Y$,*

where $K_1 = \sum_{x \in X} \mu(x)$ and $K_2 = \sum_{y \in Y} \mu'(y)$.

Intuitively, a weight function Δ shows how the probability $\mu(x)$ can be distributed among the related states y such that $\mu'(y)$ equals the total amount of probability it gets distributed by Δ. (Note that $K_1 = K_2 = 1$ for stochastic μ and μ'.) Δ is a probability distribution on $X \times Y$ such that the probability to select (x,y) with $x \sqsubseteq y$ is one. In addition, the probability to select an element in $\sqsubseteq$ whose first component is x equals $\mu(x)$, and the probability to select an element in $\sqsubseteq$ whose second component is y equals $\mu'(y)$.

Example 1. Let $X = \{ s,t \}$ and $Y = \{ u,v,w \}$ with $\mu(s) = \frac{2}{9}$, $\mu(t) = \frac{2}{3}$ and $\mu'(u) = \frac{1}{3}$, $\mu'(v) = \frac{4}{9}$ and $\mu'(w) = \frac{1}{9}$; $K_1 = K_2 = \frac{8}{9}$. Note that μ and μ' are both sub-stochastic. Let $\sqsubseteq\, = (X \times Y) \setminus \{ (s,w) \}$. We have $\mu \preceq \mu'$, as weight function Δ defined by $\Delta(s,u) = \Delta(s,v) = \Delta(t,w) = \frac{1}{8}$, $\Delta(t,v) = \frac{3}{8}$ and $\Delta(t,u) = \frac{1}{4}$ satisfies the constraints of Def. 3.

For fully probabilistic systems we consider a slight variant of probabilistic simulation by Jonsson and Larsen [26]:

Definition 4. *For FPS $(S, \mathbf{P}, L)$, let $\sqsubseteq_p$ be the coarsest binary relation on the state space S such that for all $s_1 \sqsubseteq_p s_2$:*

1. *$L(s_1) = L(s_2)$ and*
2. *$\mathbf{P}(s_1, \cdot) \preceq \mathbf{P}(s_2, \cdot)$, where $\preceq$ uses $\sqsubseteq_p$*

Relation $\sqsubseteq_p$ is symmetric if the transition probabilities are stochastic [26]. In this case, the simulation preorder agrees with probabilistic bisimulation [30]. Thus, for instance, $\sqsubseteq_p$ and probabilistic bisimulation $\sim_p$ coincide for an embedded DTMC of a CTMC.

Simulation for CTMCs. For CTMCs we modify $\sqsubseteq_p$ such that timing aspects are incorporated. Intuitively, we intend a simulation preorder to ensure that s_2 simulates s_1 iff (i) s_2 is "faster than" s_1 and (ii) the time-abstract behaviour of s_2 simulates that of s_1. An obvious attempt in this direction would be to refine Def. 4 by demanding that in addition to $\mathbf{P}(s_1, \cdot) \preceq \mathbf{P}(s_2, \cdot)$, we have $E(s_1) \leqslant E(s_2)$, i.e., s_2 should be on average at least as fast as s_1. However, such an approach turns out not to be very useful, as this would coincide with lumping equivalence [13] for uniformised CTMCs. Therefore, we present a more involved definition, which – in return – also enables a more radical state-space aggregation, since it incorporates a notion of *stuttering* [12, 20].

Definition 5. *Let $\mathcal{M} = (S, \mathbf{R}, L)$ be a CTMC. Relation $\sqsubseteq \subseteq S \times S$ is a* simulation *iff for all states $s_1, s_2 \in S$ with $s_1 \sqsubseteq s_2$ we have that $L(s_1) = L(s_2)$ and there exist functions $\Delta : S \times S \to [0,1]$, $\delta_i : S \to [0,1]$ and sets $U_i, V_i \subseteq S$ (for $i = 1, 2$) with:*

$$U_i = \{\, u_i \in S \mid \mathbf{R}(s_i, u_i) > 0 \wedge \delta_i(u_i) > 0 \,\} \text{ and}$$
$$V_i = \{\, v_i \in S \mid \mathbf{R}(s_i, v_i) > 0 \wedge \delta_i(v_i) < 1 \,\}$$

such that:

1. *$v_1 \sqsubseteq s_2$ for any $v_1 \in V_1$ and $s_1 \sqsubseteq v_2$ for any $v_2 \in V_2$,*
2. *$\Delta(u_1, u_2) > 0$ implies $u_1 \in U_1$, $u_2 \in U_2$ and $u_1 \sqsubseteq u_2$,*
3. *$K_1 \cdot \sum_{u_2 \in U_2} \Delta(w, u_2) = \delta_1(w) \cdot \mathbf{P}(s_1, w)$ and*
   $K_2 \cdot \sum_{u_1 \in U_1} \Delta(u_1, w) = \delta_2(w) \cdot \mathbf{P}(s_2, w)$, for all $w \in S$, and
4. *$K_1 \cdot E(s_1) \leqslant K_2 \cdot E(s_2)$*

where $K_i = \sum_{u_i \in U_i} \delta_i(u_i) \cdot \mathbf{P}(s_i, u_i)$ for $i = 1, 2$.

Definition 6. *The simulation relation $\sqsubseteq_m$ is defined by: $s_1 \sqsubseteq_m s_2$ iff there exists a simulation $\sqsubseteq$ such that $s_1 \sqsubseteq s_2$.*

The successor states of s_i are grouped into the subsets U_i and V_i. Although we do not require that U_i and V_i are disjoint, to understand the definition first consider $U_i \cap V_i = \varnothing$. (The fact that we allow a non-empty intersection has technical reasons that will be explained later). K_i denotes the total probability to move from s_i within one transition to a state in U_i. Vice versa, with probability $1-K_i$, in state s_i a transition to some state in V_i is made (cf. Fig. 1). The first

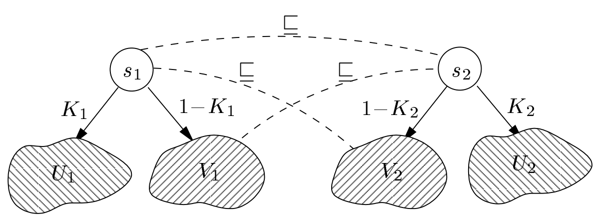


Fig. 1. Simulation scenario

condition states that the grouping of successor states into V_i and U_i is such that any state in V_2 simulates s_1 and that s_2 simulates any state in V_1. Intuitively, we interpret the moves from s_i to a V_i-state as silent transitions (i.e., a τ-transition for action-labeled transition systems). The first condition thus guarantees that any such transition is a "stutter" step. The second and third condition require the existence of a weight function Δ that relates the conditional probabilities to move from s_1 to a U_1-state and the conditional probabilities for s_2 to move to a U_2-state. Thus, Δ is a weight function for the probability distributions $\delta_i(\cdot) \cdot \mathbf{P}(s_i, \cdot)/K_i$. Intuitively, the transitions from s_i to a U_i-state are considered as observable moves and the second and third condition are the continuous versions of similar conditions for strong simulation ($\sqsubseteq_p$) in the discrete-time case. Finally, the fourth condition states that s_2 is "faster than" s_1 in the sense that the total rate to move from s_2 to a U_2-state is at least the total rate to move from s_1 to a U_1-state.

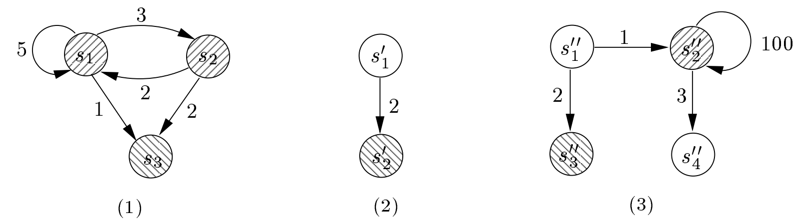


Fig. 2. Some examples of simulation refinement: $s_1 \sqsubseteq_m s'_1$ and $s'_1 \sqsubseteq_m s''_1$

Example 2. Consider the three CTMCs depicted in Fig. 2 where states s_1, s_2, s'_1, s''_1 and s''_2 are labelled with proposition a, and the other states by b. We have $s_1 \sqsubseteq_m s'_1$, since there exists a relation $\sqsubseteq = \{ (s_1, s'_1), (s_3, s'_2), (s'_2, s_3), (s_2, s'_1) \}$ with $U_1 = \{ s_3 \}, V_1 = \{ s_1, s_2 \}$, $\delta_1(s_3) = 1$ and 0 otherwise, $U_2 = \{ s'_2 \}$, $V_2 = \varnothing$, $\delta_2(s'_2) = 1$ and 0 otherwise, and $\Delta(s_3, s'_2) = \Delta(s'_2, s_3) = 1$ and 0 otherwise. It follows that $K_1 = \frac{1}{9}$ and $K_2 = 1$. (In the pictorial representation, the elements of U_i and V_i are indicated by the same patterns used in Fig. 1 for U_i and V_i). It is not difficult to check that indeed all constraints of Def. 5 are fulfilled, e.g., for the fourth constraint we obtain $\frac{1}{9} \cdot 9 \leqslant 1 \cdot 2$. Note that $s_1 \not\sqsubseteq_m s_2$ if $\mathbf{R}(s_2, s_3) > 2$ (rather than being equal to 2), since then $s_2 \sqsubseteq s'_1$ can no longer be established.

We further have $s'_1 \sqsubseteq_m s''_1$ since there exists a relation $\sqsubseteq = \{ (s'_1, s''_1), (s'_1, s''_2), (s'_2, s''_3), (s''_3, s'_2), (s'_2, s''_4), (s''_4, s'_2) \}$ with $U_1 = \{ s'_2 \}$, $V_1 = \varnothing$, $K_1 = 1$, and $\delta_1(s'_2) = 1$ and 0 otherwise, $U_2 = \{ s''_3 \}$, $V_2 = \{ s''_2 \}$, $\delta_2(s''_3) = 1$ and 0 otherwise, $K_2 = \frac{2}{3}$ and $\Delta(s''_3, s'_2) = \Delta(s'_2, s''_3) = 1$. It is straightforward to check that indeed all constraints of Def. 5 are fulfilled.

In the examples so far, we have used the special case where $\delta_i(s) \in \{ 0, 1 \}$ for any state s. In this case, δ_i is the characteristic function of U_i, and the sets U_i and V_i are disjoint. In general, though, things are more complicated and we need to construct U_i and V_i using *fragments* of states. That is, we deal with functions δ_i where $0 \leqslant \delta_i(s) \leqslant 1$. Intuitively, the $\delta_i(s)$-fragment of state s belongs to U_i, while the remaining part (the $(1-\delta_i(s))$-part) of s belongs to V_i. The use of fragments of states is exemplified in the following example.

Example 3. Consider the two CTMCs depicted in Fig. 3. where $L(s_1) = L(s_3) = L(s'_1) = L(s'_3) = \{ a \}$; the other states are labelled by b. Intuitively, s_1 is "slower than" s'_1. However, when we require the sets U_i, V_i in Def. 5 to be disjoint, then $s_1 \not\sqsubseteq_m s'_1$. This can be seen as follows. We have $s_1 \not\sqsubseteq_m s'_3$ (and hence, $V_2 = \varnothing$) as s_1 moves with rate 1 to a b-state while the total rate for s'_3 to move to a b-state is smaller (i.e., $\frac{1}{2}$). Hence, the only chance to define the components in Def. 5 is $V_2 = \varnothing$ and $U_2 = \{ s'_2, s'_3 \}$. Because s'_3 and s_2 are not comparable with the simulation order (as they have different labels), we would have to define

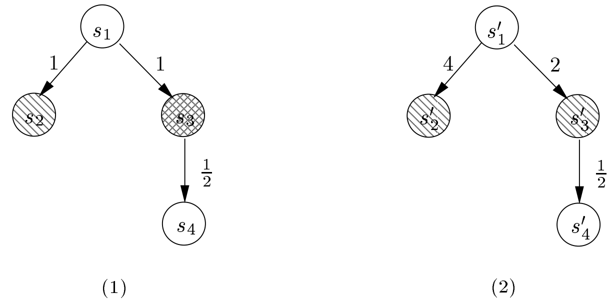


Fig. 3. An example of simulation using fragments of states

$U_1 = \{ s_2, s_3 \}$ and $V_1 = \varnothing$. But then, the weight-function condition is violated because s_1 moves with probability $\frac{1}{2}$ to a b-state while the probability for s'_1 to reach a b-state within one step is $\frac{2}{3}$.

On the other hand, when we allow s_3 to be split: one half of s_3 belongs to U_1, one half to V_1, i.e., $\delta_1(s_3) = \frac{1}{2}$ and $U_1 = \{ s_2, s_3 \}$, $V_1 = \{s_3\}$ then we get that with $U_2 = \{ s'_2, s'_3 \}$, $V_2 = \varnothing$ and $\sqsubseteq_m = \{ (s_1, s'_1), (s_2, s'_2), (s_3, s'_1), (s_4, s'_4), (s_2, s'_4) \}$ the conditional probabilities for the U_i-states are related via $\preceq$. Note that $K_1 = \frac{1}{2} + \frac{1}{4} = \frac{3}{4}$, $K_2 = 1$ and $\Delta(s_2, s'_2) = \frac{2}{3}$, $\Delta(s_3, s'_3) = \frac{1}{3}$.

Remark 1. It is interesting to observe what happens if $s_1 \sqsubseteq_m s_2$ and one of the states is absorbing. If s_2 is absorbing (i.e., $E(s_2) = 0$) then $K_1 \cdot E(s_1) = 0$. Hence, either s_1 has to be absorbing or $K_1 = 0$. In the latter case, we have $\delta_1(u_1) = 0$ for all $u_1 \in U_1$ (by condition 3. in Def. 5), i.e., all successor states of s_1 belong to V_1 and are simulated by s_2 (by condition 2. in Def. 5). Vice versa, for any state $u_2 \in U_2$:

$$0 \ < \ \delta_2(u_2) \cdot \mathbf{P}(s_2, u_2) \ = \ \sum_{u_1 \in U_1} \Delta(u_1, u_2).$$

Thus, $\Delta(u_1, u_2) > 0$ for some $u_1 \in U_1$. In particular, if $U_2 \neq \varnothing$ then $U_1 \neq \varnothing$, which implies that s_1 is non-absorbing. This shows that, if s_1 is absorbing then all successor states of s_2 belong to V_2 and simulate s_1 (by condition 2. of Def. 5).

The observation that an absorbing state s_1 is simulated by any state s_2 with the same labeling is natural for any type of simulation that abstracts from silent moves. The observation that any state s_2 which simulates an absorbing state s_1 can only perform stutter steps (non-observable transitions) can be viewed as the probabilistic counterpart to divergence for non-probabilistic systems. Note that in absorbing states of a CTMC just time advances.

Lemma 1. *$\sqsubseteq_m$ is a preorder.*

Lemma 2. *For CTMC $\mathcal{M} = (S, \mathbf{R}, L)$ and $s_1, s_2 \in S$ we have:*

$$s_1 \sqsubseteq_m^{\mathcal{M}} s_2 \textit{ if and only if } s_1 \sqsubseteq_m^{\mathit{unif}(\mathcal{M})} s_2.$$

Here, the superscript of the simulation preorder indicates the CTMC on which it is considered. The proofs of these facts are in [9]; we note here that the proof of Lemma 2 relies on the fact that sets U_i and V_i may overlap.

3 Safe and Live CSL

This section recapitulates the logic CSL and discusses two distinguished subsets of the logic that will in the sequel be shown to be weakly preserved by our simulation.

Paths in CTMCs. A path through a CTMC is an alternating sequence $\sigma = s_0\, t_0\, s_1\, t_1\, s_2 \ldots$ with $\mathbf{R}(s_i, s_{i+1}) > 0$ and $t_i \in \mathbb{R}_{>0}$ for all i.[2] The time stamps t_i denote the amount of time spent in state s_i. Let $Path$ denote the set of paths through $\mathcal{M}$. $\sigma[i]$ denotes the $(i{+}1)$th state of σ, i.e. $\sigma[i] = s_{i+1}$. $\sigma@t$ denotes the state of σ occupied at time t, i.e. $\sigma@t = \sigma[i]$ with i the smallest index such that $t < \sum_{j=0}^{i} t_j$. Let $\Pr_s$ denote the unique probability measure on sets of paths that start in s (for a definition of the Borel space see [8]).

Continuous Stochastic Logic. CSL [8] is a branching-time temporal logic à la CTL where the state-formulas are interpreted over states of a CTMC and the path-formulas are interpreted over paths in a CTMC. CSL is a variant of the (equally named) logic by Aziz *et al.* [3] and incorporates (i) an operator to refer to the probability of the occurrence of particular paths, similar to PCTL [22], a (ii) real-time until-operator, like in TCTL [1], and (iii) a steady-state operator [8]. In this paper, we focus on a fragment of CSL (denoted CSL$^-$), distinguished in that we do not consider the next step and steady-state operator. (For simplicity, we also only consider time-intervals of the form $[0, t]$.) The omission of these operators will be justified later on. Besides the usual strong until-operator we incorporate a weak until-operator that will be used in the classification of safety and liveness properties. These properties are subjects of the weak preservation results we aim to establish.

Recall that AP is the set of atomic propositions. Let $a \in AP$, $p \in [0, 1]$ and $\unlhd\ \in \{\leqslant, \geqslant\}$ and $t \in \mathbb{R}_{\geqslant 0}$ (or ∞). The syntax of CSL$^-$ is:

$$\Phi ::= a \;\Big|\; \Phi \wedge \Phi \;\Big|\; \neg\Phi \;\Big|\; \mathcal{P}_{\unlhd p}(\Phi\, \mathcal{U}^{\leqslant t}\, \Phi) \;\Big|\; \mathcal{P}_{\unlhd p}(\Phi\, \mathcal{W}^{\leqslant t}\, \Phi).$$

$\mathcal{P}_{\unlhd p}(\varphi)$ asserts that the probability measure of the paths satisfying φ meets the bound given by $\unlhd p$. The operator $\mathcal{P}_{\unlhd p}(.)$ replaces the usual (fair) CTL path quantifiers $\exists$ and $\forall$. The path-formula $\Phi\, \mathcal{U}^{\leqslant t}\, \Psi$ asserts that Ψ is satisfied at some time instant before t and that at all preceding time instants Φ holds (strong until). The weak until-operator $\mathcal{W}$ differs in that we do not require that Ψ eventually becomes true, i.e., $\Phi\, \mathcal{W}^{\leqslant t}\, \Psi$ means $\Phi\, \mathcal{U}^{\leqslant t}\, \Psi$ unless always Φ in the time-interval $[0, t]$ holds.

Semantics. The semantics of CSL for the boolean operators is identical to that for CTL and is omitted here. For the remaining state-formulas [8]:

$$s \models \mathcal{P}_{\unlhd p}(\varphi) \;\text{ iff }\; Prob(s, \varphi) \unlhd p$$

[2] For paths that end in an absorbing state s_k we assume a path to be represented as an infinite sequence $s_0\, t_0\, s_1 \ldots t_{k-1}\, s_k\, 1\, s_k\, 1\, s_k\, 1 \ldots$.

for path-formula φ. Here, $Prob(s, \varphi) = \Pr_s\{\,\sigma \in Path \mid \sigma \models \varphi\,\}$. The semantics of $\mathcal{U}^{\leqslant t}$ is defined by:

$$\sigma \models \Phi\,\mathcal{U}^{\leqslant t}\,\Psi \text{ iff } \exists x \leqslant t.\ (\sigma@x \models \Psi \wedge \forall y < x.\, \sigma@y \models \Phi)\ .$$

Note that the standard (i.e., untimed) until operator is obtained by taking t equal to ∞. The semantics of the weak until operator is defined by:

$$\sigma \models \Phi\,\mathcal{W}^{\leqslant t}\,\Psi \text{ iff } (\forall x \leqslant t.\, \sigma@x \models \Phi)\ \vee\ \sigma \models \Phi\,\mathcal{U}^{\leqslant t}\,\Psi.$$

The other boolean connectives are derived in the usual way, i.e., $\mathrm{tt} = a \vee \neg a$, $\mathrm{ff} = \neg\mathrm{tt}$, $\Phi_1 \vee \Phi_2 = \neg(\neg\Phi_1 \wedge \neg\Phi_2)$, and $\Phi_1 \to \Phi_2 = \neg\Phi_1 \vee \Phi_2$. Temporal operators like $\Diamond$, $\Box$ and their real-time variants $\Diamond^{\leqslant t}$ or $\Box^{\leqslant t}$ can be derived, e.g.

$$\mathcal{P}_{\trianglelefteq p}(\Diamond^{\leqslant t}\,\Phi) = \mathcal{P}_{\trianglelefteq p}(\mathrm{tt}\,\mathcal{U}^{\leqslant t}\,\Phi) \ \text{ and }\ \mathcal{P}_{\trianglelefteq p}(\Box^{\leqslant t}\,\Phi) = \mathcal{P}_{\trianglelefteq p}(\Phi\,\mathcal{W}^{\leqslant t}\,\mathrm{ff}).$$

For instance, if *error* is an atomic proposition that characterizes all states where a system error has occurred then $\mathcal{P}_{<0.001}(\Diamond^{\leqslant 4}\,error)$ asserts that the probability for a system error within 4 time units is smaller than 0.001.

The until-operator and the weak until-operator are closely related. For any state s and CSL$^-$-formula Φ and Ψ we have:

$$Prob(s, \Phi\,\mathcal{U}^{\leqslant t}\,\Psi) = 1\ -\ Prob\big(s, (\neg\Psi)\,\mathcal{W}^{\leqslant t}\,\neg(\Phi \vee \Psi)\big) \tag{1}$$

$$Prob(s, \Phi\,\mathcal{W}^{\leqslant t}\,\Psi) = 1\ -\ Prob\big(s, (\neg\Psi)\,\mathcal{U}^{\leqslant t}\,\neg(\Phi \vee \Psi)\big) \tag{2}$$

Hence, the following two formulas are equivalent:

$$\mathcal{P}_{\geqslant p}(\Phi\,\mathcal{W}^{\leqslant t}\,\Psi) \ \text{ and }\ \mathcal{P}_{\leqslant 1-p}((\neg\Psi)\,\mathcal{U}^{\leqslant t}\,\neg(\Phi \vee \Psi)).$$

A similar equivalence holds when the weak until- and the until-operator are exchanged. Note that a path satisfies $\neg((\neg\Phi)\,\mathcal{U}^{\leqslant t}\,(\neg\Psi))$ if Ψ always holds, a requirement that is released as soon as Φ becomes valid.

CSL safety and liveness properties. For the weak preservation results we distinguish between safety ("something bad never happens") and liveness ("something good will eventually happen") properties. In order to do so, negations may only be attached to atomic propositions. The syntax of CSL$_{safe}$, the set of safety formulas, is defined by:

$$\Phi ::= a \ \Big|\ \neg a \ \Big|\ \Phi \wedge \Phi \ \Big|\ \Phi \vee \Phi \ \Big|\ \mathcal{P}_{\geqslant p}(\Phi\,\mathcal{W}^{\leqslant t}\,\Phi) \ \Big|\ \mathcal{P}_{\leqslant p}(\neg\Phi\,\mathcal{U}^{\leqslant t}\,\neg\Phi).$$

An example CSL safety formula is $\mathcal{P}_{\geqslant 0.99}(\Box^{\leqslant 100}\neg error)$ expressing that with probability at least 0.99 no error will occur in the next hundred time units. The syntax of CSL$_{live}$, the set of liveness formulas, is defined by:

$$\Phi ::= a \ \Big|\ \neg a \ \Big|\ \Phi \wedge \Phi \ \Big|\ \Phi \vee \Phi \ \Big|\ \mathcal{P}_{\geqslant p}(\Phi\,\mathcal{U}^{\leqslant t}\,\Phi) \ \Big|\ \mathcal{P}_{\leqslant p}(\neg\Phi\,\mathcal{W}^{\leqslant t}\,\neg\Phi).$$

As a result of the aforementioned relationship between $\mathcal{U}$ and $\mathcal{W}$ (cf. equations (1) and (2)), there is a duality between safety and liveness properties for CSL, i.e., for any formula Φ_{safe} there is a liveness property equivalent to $\neg\Phi_{safe}$, and the same applies to liveness property Φ_{live}.

Next and steady state. Neither the next operator $\mathcal{P}_{\trianglelefteq p}(X\Phi)$, nor the steady-state operator $\mathcal{S}_{\trianglelefteq p}(\Phi)$ of [8] can become part of a CSL fragment that enables a weak preservation result for $\sqsubseteq_m$. This is shown by the following example.

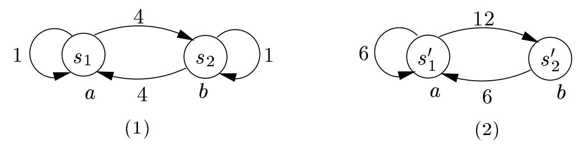


Fig. 4. Next and steady state behaviour is not preserved by $\sqsubseteq_m$.

Example 4. Consider the two CTMCs depicted in Fig. 4, where each state is decorated with the atomic propositions valid in the respective state. We have $s_1 \sqsubseteq_m s'_1$ and $s_2 \sqsubseteq_m s'_2$. The steady-state (or long-run) probability $\pi(s_i)$ of being in state s_i is spread evenly among s_1 and s_2, whereas it is spread unevenly among s'_1 and s'_2; s'_1 is less likely than s'_2. Concretely $\pi(s_1) = \pi(s_2) = \frac{1}{2}$ but $\pi(s'_1) = \frac{1}{3}$ and $\pi(s'_2) = \frac{2}{3}$. As a consequence, $s_1 \models \mathcal{S}_{\geqslant 0.5}(a)$, but $s'_1 \not\models \mathcal{S}_{\geqslant 0.5}(a)$. On the other hand, $s_2 \models \mathcal{S}_{\leqslant 0.5}(b)$, while $s'_2 \not\models \mathcal{S}_{\leqslant 0.5}(b)$. Furthermore, we have that $s_1 \models \mathcal{P}_{\leqslant 0.2}(Xa)$ and $s_2 \models \mathcal{P}_{\geqslant 0.2}(Xb)$, but $s'_1 \not\models \mathcal{P}_{\leqslant 0.2}(Xa)$ and $s'_2 \not\models \mathcal{P}_{\geqslant 0.2}(Xb)$.

The fact that the steady-state operator is not compatible with our simulation relation can be viewed as a specific instance of the well-known phenomenon that CTMCs cannot be ordered according to their steady-state performance [35, 11].

4 Weak Preservation

This section is devoted to the main result of the paper: weak preservation of the two CSL fragments CSL_{safe} and CSL_{live} with respect to $\sqsubseteq_m$. To arrive there, requires to establish some crucial observations.

For a given CTMC $\mathcal{M}$ we first remark that the probability measures on CTMC $\mathcal{M}$ agree with those on the uniformised CTMC $\mathit{unif}(\mathcal{M})$. For arbitrary CSL path-formula φ we have:

Lemma 3. $\Pr_s^{\mathcal{M}}\{\, \sigma \in \mathit{Path} \mid \sigma \models \varphi \,\} \;=\; \Pr_s^{\mathit{unif}(\mathcal{M})}\{\, \sigma \in \mathit{Path} \mid \sigma \models \varphi \,\}$.

The above lemma implies that CSL satisfaction on $\mathcal{M}$ agrees with CSL satisfaction on $\mathit{unif}(\mathcal{M})$. We thus may safely assume that the exit rate of each state equals E.

Theorem 1. *For state s_1, s_2:*

1. *for CSL_{safe}-formula Φ_{safe}: $s_1 \sqsubseteq_m s_2 \implies (s_2 \models \Phi_{safe} \implies s_1 \models \Phi_{safe})$.*
2. *for CSL_{live}-formula Φ_{live}: $s_1 \sqsubseteq_m s_2 \implies (s_2 \not\models \Phi_{live} \implies s_1 \not\models \Phi_{live})$.*

Proof. It is first proven that sets $\mathit{Sat}(\Phi_{safe})$ are upward-closed, i.e., if $s_1 \sqsubseteq_m s_2$ and $s_1 \in \mathit{Sat}(\Phi_{safe})$ then $s_2 \in \mathit{Sat}(\Phi_{safe})$. This is not involved and omitted here. The proof is then by induction on the formula, where the interesting cases ($\mathcal{U}^{\leqslant t}$ and $\mathcal{W}^{\leqslant t}$) use Lemma 4 below. The statement for the CSL_{live}-formulas follows then by duality of the weak until- and until-operator. ■

The proof of the above theorem requires to establish the following fact (Lemma 4):

$$s_1 \sqsubseteq_m s_2 \text{ implies } \mathit{Prob}(s_1, \Phi_1 \,\mathcal{U}^{\leqslant t}\, \Phi_2) \;\leqslant\; \mathit{Prob}(s_2, \Phi_1 \,\mathcal{U}^{\leqslant t}\, \Phi_2), \qquad (3)$$

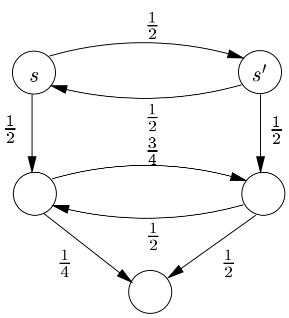
where sets $\mathit{Sat}(\Phi_i) = \{\, s \in S \mid s \models \Phi_i \,\}$ are upward-closed. The initial proof idea for this fact is to resort to the embedded uniformised CTMC of $\mathcal{M}$, using the result that:

$$\mathit{Prob}^{\mathcal{M}}(s_1, \Phi_1 \,\mathcal{U}^{\leqslant t}\, \Phi_2) = e^{-E \cdot t} \cdot \sum_{k=0}^{\infty} \frac{(E \cdot t)^k}{k!} \cdot \mathit{Prob}^{\mathcal{D}}(s_1, \Phi_1 \,\mathcal{U}^{\leqslant k}\, \Phi_2), \qquad (4)$$

where $\mathcal{D} = \mathit{emb}(\mathit{unif}(\mathcal{M}))$ and $\Phi_1 \,\mathcal{U}^{\leqslant k}\, \Phi_2$ means that Φ_2 can be reached within at most k steps via a Φ_1-path (for natural k) [22]. The advantage of this approach would be that the remaining proof obligation:

$$s_1 \sqsubseteq_m s_2 \text{ implies } \mathit{Prob}^{\mathcal{D}}(s_1, \Phi_1 \,\mathcal{U}^{\leqslant k}\, \Phi_2) \;\leqslant\; \mathit{Prob}^{\mathcal{D}}(s_2, \Phi_1 \,\mathcal{U}^{\leqslant k}\, \Phi_2), \text{ for any } k \qquad (5)$$

could be verified by considering the discrete-time behaviour of the CTMC only. Whereas the proof of equation (4) is rather straightforward, the conjecture (5) turns out to be wrong. This is illustrated by the following (uniformised) CTMC $\mathcal{M}$:



where only the absorbing state is labelled by proposition b. It is not difficult to check that state s' simulates state s. Indeed it follows that $\mathit{Prob}^{\mathcal{M}}(s, \Diamond^{\leqslant t}\, b) \leqslant \mathit{Prob}^{\mathcal{M}}(s', \Diamond^{\leqslant t}\, b)$ for any real time instant t. However, $\mathit{Prob}^{\mathrm{emb}(\mathcal{M})}(s, \Diamond^{\leqslant k}\, b) = \frac{7}{16} \not\leqslant \frac{3}{8} = \mathit{Prob}^{\mathrm{emb}(\mathcal{M})}(s', \Diamond^{\leqslant k}\, b)$ for $k = 3$. This contradicts (5). Thus, this initial proof attempt fails and we have to consider an alternative route. Alternative proof attempts along similar lines failed. We prove (3) therefore in a different way. The crux of the proof is to apply a number of transformations on the CTMC under consideration. The details of the proof are in [9]; the proof sketch is given below.

Lemma 4. *Let Φ_1 and Φ_2 be CSL-formulas such that the satisfaction sets $\mathit{Sat}(\Phi_i)$ are upward-closed, i.e., if $s_1 \sqsubseteq_m s_2$ and $s_1 \in \mathit{Sat}(\Phi_i)$ then $s_2 \in \mathit{Sat}(\Phi_i)$ for $i = 1, 2$. Then:*

$$s_1 \;\sqsubseteq_m\; s_2 \textit{ implies } \mathit{Prob}(s_1, \Phi_1 \,\mathcal{U}^{\leqslant t}\, \Phi_2) \;\leqslant\; \mathit{Prob}(s_2, \Phi_1 \,\mathcal{U}^{\leqslant t}\, \Phi_2).$$

Proof. We only provide the proof sketch here; the full proof is given in [9]. Through a series of transformation steps we modify $\mathcal{M}$ to obtain a CTMC such that for any pair $s_1 \sqsubseteq_m s_2$:

- The probability to move from s_1 to a V_1-state equals the probability for the added self-loop $s_2 \to s_2$.
- The probability for the added self-loop $s_1 \to s_1$ equals the probability to move from s_2 to a V_2-state.
- The probabilities to move from s_1 and s_2 to a U_1- and U_2-state, respectively, are equal.
- s_2 is faster than s_1, i.e., the exit rate of s_2 exceeds the exit rate of s_1.

(The meaning of U_1, U_2, V_1 and V_2 is as in Def. 5.) The reasoning will then be as follows. The interesting case is $s_i \models \Phi_1 \wedge \neg\Phi_2$ for $i = 1, 2$. Hence, all states in V_1 and V_2 satisfy Φ_1 but not Φ_2. Thus, the only possibility for s_i to fulfill the path-formula $\Phi_1 \, \mathcal{U}^{\leqslant t} \, \Phi_2$ is to move to a U_i-state. Let $p(s, t, n)$ denote the probability for s to reach a Φ_2-state in at most t time units within at most n transitions via Φ_1-states. Then, $Prob(s_i, \Phi_1 \, \mathcal{U}^{\leqslant t} \, \Phi_2)$ equals $\lim_{n\to\infty} p(s_i, t, n)$. Via the introduction of (yet another) state s_2' that has the same probabilistic behaviour as s_2 but the exit rate of s_1 we then establish $p(s_2', t, n) \leqslant p(s_2, t, n)$. By induction on n it is subsequently shown that $p(s_1, t, n) \leqslant p(s_2', t, n)$. ■

5 Simulation Equivalence

This section defines simulation equivalence ($\equiv_m$) and relates this notion to the equivalences induced by the two CSL fragments. Furthermore, the relationship with lumping equivalence [13], probabilistic (bi)simulation [30, 26] and weak probabilistic bisimulation [7] is established.

Simulation equivalence. Simulation equivalence denotes the kernel of the simulation preorder. Two states are simulation equivalent if and only if they are mutually simulating each other:

Definition 7. *$s_1 \equiv_m s_2$ if and only if $s_1 \sqsubseteq_m s_2$ and $s_2 \sqsubseteq_m s_1$.*

Theorem 2. *Let $s_1 \equiv_m s_2$. Then:*

1. *for any CSL safety-formula Φ_{safe}: $s_1 \models \Phi_{safe}$ iff $s_2 \models \Phi_{safe}$*
2. *for any CSL liveness-formula Φ_{live}: $s_1 \models \Phi_{live}$ iff $s_2 \models \Phi_{live}$*

Lemma 5. *CSL_{safe}-equivalence and CSL_{live}-equivalence are simulations.*

Theorem 3. *For any states s_1, s_2:*

$$s_1 \equiv_m s_2 \text{ iff } (s_1, s_2 \text{ are } CSL_{safe}\text{-equivalent}) \text{ iff } (s_1, s_2 \text{ are } CSL_{live}\text{-equivalent}).$$

Thus, simulation is characterised by each of the two fragments of CSL we considered.

Lumping equivalence. Recall from [6] that two states s_1 and s_2 are lumping equivalent ($s_1 \sim_m s_2$) if there is some equivalence relation R on S with $(s_1, s_2) \in R$ satisfying that whenever $(s, s') \in R$ then $L(s) = L(s')$ and for all equivalence classes C in the quotient S/R,

$$\sum_{s'' \in C} \mathbf{R}(s, s'') \;=\; \sum_{s'' \in C} \mathbf{R}(s', s'').$$

Theorem 4. *For any state s_1, s_2: $s_1 \sim_m s_2$ implies $s_1 \equiv_m s_2$.*

The converse of the above theorem does not hold. For instance, two corresponding states in a CTMC $\mathcal{M}$ and $\mathit{unif}(\mathcal{M})$ simulate each other (if considered in the disjoint union of the state spaces), but are not lumping equivalent if the uniformisation rate E is chosen strictly larger than $max_{s \in S} E(s)$. Thus simulation equivalence strictly refines lumping equivalence.

Simulation on DTMCs. It is interesting to investigate the effect of our simulation relation if interpreted without the constraint on the total rates of states, i.e., on (embedded) DTMCs. For a given DTMC $(S, \mathbf{P}, L)$, let $\leqslant_p$ be the preorder obtained by omitting clause *4.* from Def. 5, and let $\equiv_p$ denote the induced simulation equivalence (cf. Def. 7). We have that strong probabilistic bisimulation ($\sim_p$) is finer than $\equiv_p$, and so is weak probabilistic bisimulation [7]: Let $\approx_p$ denote (state-labelled) weak probabilistic bisimulation. More specific, two states s_1 and s_2 are weakly probabilistic bisimilar ($s_1 \approx_p s_2$) if there is some equivalence relation R on S with $(s_1, s_2) \in R$ satisfying that whenever $(s, s') \in R$ then $L(s) = L(s')$ and for all equivalence classes C in the quotient S/R,

$$\mathbf{W}(s, C) \;=\; \mathbf{W}(s', C)$$

where $\mathbf{W}(s, C) = \sum_{s'' \in [s]_R} \mathbf{P}(s, s'')\mathbf{W}(s'', C)$ if $s \notin C$, and 1 otherwise ($[s]_R$ is the equivalence class of R containing s).[3]

Theorem 5. *For any state s_1, s_2 of a DTMC: $s_1 \approx_p s_2$ implies $s_1 \equiv_p s_2$.*

We claim that the converse direction of this theorem holds as well in the DTMC setting (but not for FPSs) though we have not formally shown this yet. Recall that $\sqsubseteq_p$ and $\sim_p$ agree on DTMCs, and we feel that a similar result may be expected for $\leqslant_p$ and $\approx_p$. Note that the probabilistic preorder is a side issue of our work since we are mainly interested in CTMC model checking.

6 Related Work

Preservation and bisimulation. Aziz *et al.* [2] have shown that Larsen-Skou probabilistic bisimulation [30] on discrete-time Markov chains fully preserves any formula in the logic Probabilistic CTL (PCTL) [22]. This result has recently been

[3] Here we define $\approx_p$ using the branching bisimulation style, see [7] for a proof that both styles coincide on DTMCs.

generalised towards continuous-space Markov processes by Desharnais *et al.* [17]. Segala and Lynch [34] reported similar results for simple probabilistic automata, a model in which probabilistic choices and non-determinism co-exist. Baier *et al.* [6] have shown that lumping equivalence [13] preserves CSL; Desharnais and Panangaden [18] have recently shown the converse, namely that the equivalence induced by CSL implies lumping equivalence.

Simulation preorders. Based on the seminal works by Larsen and Skou [30] and Jonsson and Larsen [26] on probabilistic (bi)simulation several variants have been proposed, see e.g., [34, 7, 10, 32, 37]. Mostly related to this paper are the simulations of [16, 34, 19]. We discuss these works briefly.

D'Argenio *et al.* [16] investigated simulation on discrete-time Markov decision processes, and showed preservation of (untimed) probabilistic reachability properties. Opposed to their work, our approach stays in an entirely probabilistic setting – we do not abstract away probabilistic behaviour. This has the advantage that CSL model-checking algorithms can be applied to the abstract model as well as to the concrete model.

Segala and Lynch [34] presented weak and strong simulations for action-labelled probabilistic automata and showed that these notions are pre-congruences wrt. parallel composition. For divergence-free probabilistic automata they showed that strong simulation weakly preserves a "safe" fragment of PCTL [22]. In addition, a weak preservation result for weak simulation for a fragment of (a subset of) a variant of PCTL that abstracts from internal activities is shown.

Desharnais *et al.* [19] studied the approximation of continuous-space Markov processes by a series of finite (rational) Markov processes. They used a simulation preorder to capture the relationship between successive finite approximants and showed that this preorder weakly preserves a subset of PML, a probabilistic variant of Hennessy-Milner logic.

Testing preorders. Another important branch of preorders are the ones based on testing, a framework in which processes are compared by their (in)ability to pass a specified set of tests. For discrete-time probabilistic systems, a whole range of testing preorders have been proposed. A recent account can be found in [27] where also the relation between probabilistic may-testing and probabilistic simulation is established. Testing preorders for continuous-time probabilistic systems have received scant attention so far. A notable exception is the work by Bernardo and Cleaveland [11] who consider testing of action-labelled CTMCs. Similar to our simulation preorder, their tests allow one to discriminate models with respect to their transient evolution. To be more precise, two testing preorders are considered, one based on the probability of executing a successful computation whose average duration is not exceeding a time bound, and one based on the probability to reach success within a time bound. It is shown that these testing preorders coincide. CSL preservation results for testing are not known to us.

7 Concluding Remarks

This paper presented a simulation preorder ($\sqsubseteq_m$) for CTMCs and provided weak preservation results for safety- and liveness-fragments of CSL. We claim that the simulation preorder can be easily extended towards Markov reward models (by requiring that rewards of simulating states are related according to $\leqslant$) and that weak preservation results for fragments of the logic CSRL [5] can be obtained in a similar way as shown in this paper. As a next step, we plan to work on an algorithm for deciding $\sqsubseteq_m$ and to construct the quotient space w.r.t. simulation preorder or simulation equivalence, based on [7, 10, 4, 32]. Moreover, we will investigate whether the concept of simulation can help to increase the efficiency of CSL model checking using an abstraction refinement methodology as in [16].

References

1. R. Alur, C. Courcoubetis and D. Dill. Model-checking in dense real-time. *Inf. and Comp.*, **104**(1): 2–34, 1993.
2. A. Aziz, V. Singhal, F. Balarin, R. Brayton and A. Sangiovanni-Vincentelli. It usually works: the temporal logic of stochastic systems. In P. Wolper (ed), *Computer-Aided Verification*, LNCS 939, pp. 155–165, 1995.
3. A. Aziz, K. Sanwal, V. Singhal and R. Brayton. Verifying continuous time Markov chains. In R. Alur and T.A. Henzinger (eds), *Computer-Aided Verification*, LNCS 1102, pp. 269–276, 1996.
4. C. Baier, B. Engelen, and M. Majster-Cederbaum. Deciding bisimilarity and similarity for probabilistic processes. *J. of Comp. and System Sc.*, **60**(1):187–231, 2000.
5. C. Baier, B.R. Haverkort, H. Hermanns and J.-P. Katoen. On the logical characterisation of performability properties. In U. Montanari *et al.* (eds.), *Automata, Languages, and Programming*, LNCS 1853, pp. 780–792, 2000.
6. C. Baier, B.R. Haverkort, H. Hermanns and J.-P. Katoen. Model checking continuous-time Markov chains by transient analysis. In E.A Emerson and A.P. Sistla (eds), *Computer-Aided Verification*, LNCS 1855, pp. 358–372, 2000.
7. C. Baier, H. Hermanns. Weak bisimulation for fully probabilistic processes. In O. Grumberg (ed), *Computer-Aided Verification*, LNCS 1254, pp. 119-130. 1997.
8. C. Baier, J.-P. Katoen and H. Hermanns. Approximate symbolic model checking of continuous-time Markov chains. In J.C.M. Baeten and S. Mauw (eds), *Concurrency Theory*, LNCS 1664, pp. 146–162, 1999.
9. C. Baier, J.-P. Katoen, H. Hermanns and B. Haverkort. Simulation for continuous-time Markov chains. Technical report, Univ. Twente, 2002.
10. C. Baier and M.I.A. Stoelinga. Norm functions for probabilistic bisimulations with delays. In J. Tyurin (ed), *Found. of Software Science and Computation Structures*, LNCS 1784, pp. 1-16. 2000.
11. M. Bernardo and R. Cleaveland. A theory of testing for Markovian processes. In C. Palamidessi (ed), *Concurrency Theory*, LNCS 1877, pp. 305–319, 2000.
12. M. Brown, E. Clarke, O. Grumberg. Characterizing finite Kripke structures in propositional temporal logic. *Th. Comp. Sc.*, **59**: 115–131, 1988.
13. P. Buchholz. Exact and ordinary lumpability in finite Markov chains. *J. of Appl. Prob.*, **31**: 59–75, 1994.

14. E. Clarke, O. Grumberg and D.E. Long. Model checking and abstraction. *ACM Tr. on Progr. Lang. and Sys.*, **16**(5): 1512–1542, 1994.
15. E. Clarke, O. Grumberg and D. Peled. *Model Checking*. MIT Press, 1999.
16. P.R. D'Argenio, B. Jeannet, H.E. Jensen, and K.G. Larsen. Reachability analysis of probabilistic systems by successive refinements. In L. de Alfaro and S. Gilmore (eds), *Process Algebra and Probabilistic Methods*, LNCS 2165, pp. 39–56, 2001.
17. J. Desharnais, A. Edalat and P. Panangaden. A logical characterisation of bisimulation for labelled Markov processes. In *IEEE Symp. on Logic in Computer Science*, pp. 478–487, 1998.
18. J. Desharnais and P. Panangaden. Continuous stochastic logic characterizes bisimulation of continuous-time Markov processes, 2001 (submitted for publication). (available at `http://www-acaps.cs.mcgill.ca/~prakash/csl.ps`).
19. J. Desharnais, V. Gupta, R. Jagadeesan and P. Panangaden. Approximating labelled Markov processes. In *IEEE Symp. on Logic in Computer Science*, pp. 95–106, 2000.
20. R.J. van Glabbeek and W.P. Weijland. Branching time and abstraction in bisimulation semantics. *J. ACM*, **43**(3): 555–600, 1996.
21. D. Gross and D.R. Miller. The randomization technique as a modeling tool and solution procedure for transient Markov chains. *Oper. Res.* **32**(2): 343–361, 1984.
22. H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *Form. Asp. of Comp.* **6**: 512–535, 1994.
23. H. Hermanns, J.-P. Katoen, J. Meyer-Kayser and M. Siegle. A Markov chain model checker. In S. Graf and M. Schwartzbach (eds), *Tools and Algs. for the Construction and Analysis of Systems*, LNCS 1785, pp. 347–362, 2000.
24. A. Jensen. Markov chains as an aid in the study of Markov processes. *Skand. Aktuarietidskrift* **3**: 87–91, 1953.
25. B. Jonsson. Simulations between specifications of distributed systems. In J.C.M. Baeten and J.F. Groote (eds), *Concurrency Theory*, LNCS 527, pp. 346–360, 1991.
26. B. Jonsson and K.G. Larsen. Specification and refinement of probabilistic processes. In *IEEE Symp. on Logic in Computer Science*, pp. 266-277, 1991.
27. B. Jonsson, W. Yi and K.G. Larsen. Probabilistic extensions of process algebras. In J. Bergstra *et al.* (eds), *Handbook of Process Algebra*, Ch. 11, pp. 685–709, 2001.
28. J.-P. Katoen, M.Z. Kwiatkowska, G. Norman and D. Parker. Faster and symbolic CTMC model checking. In L. de Alfaro and S. Gilmore (eds), *Process Algebra and Probabilistic Methods*, LNCS 2165, pp. 23–38, 2001.
29. V.G. Kulkarni. *Modeling and Analysis of Stochastic Systems*. Chapman & Hall, 1995.
30. K.G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Inf. and Comp.*, **94**(1): 1–28, 1992.
31. R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.
32. A. Philippou, I. Lee, and O. Sokolsky. Weak bisimulation for probabilistic systems. In C. Palamidessi (ed), *Concurrency Theory*, LNCS 1877, pp. 334–349, 2000.
33. M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.
34. R. Segala and N.A. Lynch. Probabilistic simulations for probabilistic processes. *Nordic J. of Computing*, **2**(2): 250–273, 1995.
35. M. Silva. Private communication. 1993.
36. W.J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton Univ. Press, 1994.
37. M.I.A. Stoelinga. *Verification of Probabilistic, Real-Time and Parametric Systems*. PhD Thesis, University of Nijmegen, 2002.