Model Checking Algorithms for CTMDPs

Peter Buchholz[1], Ernst Moritz Hahn[2], Holger Hermanns[2], and Lijun Zhang[3]

[1] Technical University of Dortmund, Computer Science, Germany
[2] Saarland University, Computer Science, Germany
[3] Technical University of Denmark, DTU Informatics, Denmark

Abstract. Continuous Stochastic Logic (CSL) can be interpreted over continuous-time Markov decision processes (CTMDPs) to specify quantitative properties of stochastic systems that allow some external control. Model checking CSL formulae over CTMDPs requires then the computation of optimal control strategies to prove or disprove a formula. The paper presents a conservative extension of CSL over CTMDPs—with rewards—and exploits established results for CTMDPs for model checking CSL. A new numerical approach based on uniformization is devised to compute time bounded reachability results for time dependent control strategies. Experimental evidence is given showing the efficiency of the approach.

1 Introduction

Model checking of continuous-time Markov chains (CTMCs) is a well established approach to prove or disprove quantitative properties for a wide variety of systems [1, 2]. If the system can be controlled by some external entity, then continuous-time Markov decision processes (CTMDPs) [3, 4] rather than CTMCs are the natural extension to be used for modeling, possibly enriched with rewards.

In this paper we formulate the model checking problem of the logic CSL—with reward extensions—in terms of decision problems in CTMDPs. The most challenging model checking subproblem for this logic is to compute the minimum/maximum reward with which a CSL formula holds. The problem contains as a specific case the problem of computing the time or time-interval bounded reachability probability in CTMDPs, a problem that has received considerable attention recently [5–10].

We introduce a numerical algorithm based on uniformization to compute, and approximate, the minimum/maximum *gain vector* per state (can be interpreted as rewards and/or costs) for a finite interval $[0, T]$ that is the key for model checking CSL formulae. The method we present is an adaption and extension of a recent algorithm [11] to compute the accumulated reward in a CTMDP over a finite interval. It works in a backward manner by starting with some initial gain vector $\mathbf{g}_T$ at time $t = T$, then it determines the optimal decision at t, and then assumes that the optimal decision is deterministic for a small interval $(t', t]$. The gain vector can then be computed for the whole interval. Afterwards, the optimal action at t' is determined, and the procedure is repeated until we arrive at $t = 0$. The correctness follows from the celebrated result by Miller [12] showing that an optimal policy exists, and only a finite number of switches of the actions is needed for describing it. It returns a control strategy that maximizes or minimizes a reward measure over a finite or an infinite time horizon.

G. Gopalakrishnan and S. Qadeer (Eds.): CAV 2011, LNCS 6806, pp. 225–242, 2011.

If reward values are *zero*, and we have the appropriate *initial value* for the gain vector $\mathbf{g}_T$, the problem can be exploited to arrive at an uniformization based approach for the computation of time bounded reachability probabilities within time T. It can easily be generalized to the maximal reachability for a finite interval $[t_0, T]$, which is the key element of checking the probabilistic operator in CSL. Moreover, by computing the gain vector between $[t_0, T]$ with $t_0 > 0$, and followed by a probabilistic reachability analysis for the interval $[0, t_0]$, we are able to compute the minimum/maximum gain vector for $[t_0, T]$: this gives us then a complete CSL model checking algorithm for CTMDPs.

Contribution. This paper provides a full CSL model checking algorithm for CTMDPs with rewards. We show that the problem, for both probabilistic operator and various reward properties, can be reduced to the computation of accumulated rewards within time T, which allows us to exploit a deep insight by Miller [12]. This then provides both theoretical and practical insights: (i) on the theoretical side, we have that all maximal (or minimal) values arising in model checking can be obtained by finite memory policies, (ii) on the practical side, we exploit recent algorithmic advances [11] to arrive at an efficient approximation algorithm—providing upper and lower bounds—based on the well known notion of uniformization. We also provide experimental evidence showing the efficiency of the new numerical approach. The improvements over the state-of-the-art are dramatic, and resemble the milestones in approximate CTMC model checking research, which was initially resorting to discretization [13], but got effective—and mainstream technology—only through the use of uniformization [2].

Organization of the paper. Section 2 provides the basic definitions. Section 3 introduces the logic CSL and shows how CSL formulae can be interpreted in terms of minimal/maximal rewards gained in CTMDPs. Afterwards, in Section 4, the basic model checking approach is presented. The key step of model checking is the computation of an appropriate gain vector. Section 5 introduces a new algorithm based on uniformization to compute the gain vector. Then the performance of the new model checking algorithm is evaluated by means of some examples in Section 6. Section 7 discusses related work, and the paper is concluded in Section 8.

2 Basic Definitions

In this section we define CTMDPs as our basic model class and formulate the general problem of computing maximal/minimal instantaneous and accumulated rewards. The following notations are mainly taken from [12] and are used similarly in [11].

Definition 1 (CTMDP). *A* continuous-time Markov decision process (CTMDP) *is a tuple* $\mathcal{C} = (\mathcal{S}, \mathcal{D}, \mathbf{Q}^{\mathbf{d}})$ *where*

- $\mathcal{S} = \{1, \ldots, n\}$ *is a finite set of states,*
- $\mathcal{D} = \bigtimes_{s=1}^{n} \mathcal{D}_s$ *where* $\mathcal{D}_s$ *is a finite set of decisions that can be taken in state* $s \in \mathcal{S}$*,*
- $\mathbf{Q}^{\mathbf{d}}$ *is an* $n \times n$ *generator matrix of a continuous-time Markov chain for each decision vector* $\mathbf{d}$ *of length* n *with* $\mathbf{d}(s) \in \mathcal{D}_s$*.*

A CTMDP with reward *is a pair* $(\mathcal{C}, \mathbf{r})$ *where* $\mathcal{C}$ *is a CTMDP and* $\mathbf{r}$ *is a nonnegative (column) reward vector of length* n*.*

Sometimes we additionally define the initial distribution $\mathbf{p}_0$ of a CTMDP, which is a row vector of length n that defines a probability distribution over the set of states $\mathcal{S}$.

We consider a time interval $[0, T]$ with $T > 0$. Let Ω denote the set of all (right continuous) step functions on $[0, T]$ into $\mathcal{S}$, and let $\mathcal{F}$ denote the σ-algebra [12] of the sets in the space Ω generated by the sets $\{\omega \in \Omega \mid \omega(t) = s\}$ for all $t \leq T$ and $s \in \mathcal{S}$.

The notation $\mathbf{d} \in \mathcal{D}$, or the variant with an index, is used for decision vectors. A *policy* π (also known as *scheduler* or *adversary*) is a mapping from $[0, T]$ into $\mathcal{D}$, and $\mathbf{d}_t$ is the corresponding decision vector at time $t \in [0, T]$, i.e., $\mathbf{d}_t(s)$ is the decision taken if the system is in state s at time t. We require that π is a measurable function where measurable means Lebesgue measurable [12, 14]. For a measurable policy π, the CTMDP with initial distribution $\mathbf{p}_0$ induces the *probability space* $(\Omega, \mathcal{F}, P^\pi_{\mathbf{p}_0})$. If we have an initial state s (i.e. $\mathbf{p}_0(s) = 1$), we write P^π_s instead of $P^\pi_{\mathbf{p}_0}$.

Let $\mathcal{M}$ be the set of all measurable policies on $[0, T]$. A policy π is *piecewise constant* if there exist some $m < \infty$ and $0 = t_0 < t_1 < t_2 < \ldots < t_{m-1} < t_m = T < \infty$ such that $\mathbf{d}_t = \mathbf{d}_{t'}$ for $t, t' \in (t_k, t_{k+1}]$ $(0 \leq k < m)$. The policy is *stationary* if $m = 1$.

For a given policy $\pi \in \mathcal{M}$, define a matrix $\mathbf{V}^\pi_{t,u}$ with $0 \leq t \leq u \leq T$ by the following differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}u}\mathbf{V}^\pi_{t,u} = \mathbf{V}^\pi_{t,u}\mathbf{Q}^{\mathbf{d}_u} \tag{1}$$

with the initial condition $\mathbf{V}^\pi_{t,t} = \mathbf{I}$. Element (i, j) of this matrix contains the probability that the CTMDP under policy π is in state j at time u when it has been in state i at time t [12]. We use the notation $\mathbf{V}^\pi_t$ for $\mathbf{V}^\pi_{0,t}$. Knowing the initial distribution $\mathbf{p}_0$ at time 0, the distribution at time t equals $\mathbf{p}^\pi_t = \mathbf{p}_0\mathbf{V}^\pi_t$ with $0 \leq t \leq T$.

Let $(\mathcal{C}, \mathbf{r})$ be a CTMDP with reward, and $G \subseteq \mathcal{S}$ a set of states of our interests. Define as $\mathbf{r}|_G$ the vector which results from assigning zero rewards to non-G states, namely $\mathbf{r}|_G(s) = \mathbf{r}(s)$ if $s \in G$ and 0 otherwise. For $t \leq T$, let $\mathbf{g}^\pi_{t,T}|_G$ be a column vector of length n defined by:

$$\mathbf{g}^\pi_{t,T}|_G = \mathbf{V}^\pi_{t,T}\mathbf{g}_T + \int_t^T \mathbf{V}^\pi_{\tau,T}\mathbf{r}|_G \,\mathrm{d}\tau \tag{2}$$

where $\mathbf{g}_T$ is the initial gain vector at time T, independent of the policies. The second part is the accumulated gain vector through G-states between $[t, T]$. Intuitively, it contains in position $s \in \mathcal{S}$ the expected reward accumulated until time T, if the CTMDP is at time t in state s and policy π is chosen. In most of the cases, T is fixed and clear from the context, then we skip it and write $\mathbf{g}^\pi_t|_G$ instead. Moreover, $|_G$ will also be skipped in case $G = \mathcal{S}$. As we will see later, $\mathbf{g}_T$ will be initialized differently for different model checking problems but is independent of π. For a given initial vector $\mathbf{p}_0$ the expected reward under policy π equals $\mathbf{p}_0\mathbf{g}^\pi_0$.

3 Continuous Stochastic Logic

To specify quantitative properties we use a conservative extension of the logic *Continuous Stochastic Logic* (CSL) introduced in [1, 2], here interpreted over CTMDPs. We relate the model checking of CSL formulae to the computation of minimal/maximal gain vectors in CTMDPs.

3.1 CSL

Let $\mathtt{I}, \mathtt{J}$ be non-empty closed intervals on $\mathbb{R}_{\geq 0}$ with rational bounds. The syntax of the CSL formulae is defined as follows:

$$\Phi := a \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}\Phi) \mid \mathbb{S}_{\mathtt{J}}(\Phi) \mid \mathbb{I}^{t}_{\mathtt{J}}(\Phi) \mid \mathbb{C}^{\mathtt{I}}_{\mathtt{J}}(\Phi)$$

where $a \in AP$, and $t \geq 0$. We use Φ, Ψ for CSL formulae, and use the abbreviations $true = a \vee \neg a$, $\Diamond^{\mathtt{I}}(\Phi) = true\ \mathsf{U}^{\mathtt{I}}\Phi$, and $\mathbb{C}^{\leq t}_{\mathtt{J}}(\Phi)$ for $\mathbb{C}^{[0,t]}_{\mathtt{J}}(\Phi)$. We refer to $\Phi\ \mathsf{U}^{\mathtt{I}}\Psi$ as a (CSL) path formula.

Except for the rightmost two operators, this logic agrees with CSL on CTMCs [2]. It should however be noted that $\mathbb{S}_{\mathtt{J}}(\Phi)$ refers to the long-run average reward gained in Φ-states, which coincides with the CTMC interpretation of CSL for a reward structure of constant 1. The rightmost two operators are inspired by the discussion in [15]. $\mathbb{I}^{t}_{\mathtt{J}}(\Phi)$ specifies that the instantaneous reward at time t in Φ-states is in the interval $\mathtt{J}$. $\mathbb{C}^{\mathtt{I}}_{\mathtt{J}}(\Phi)$ in turn accumulates (that is, integrates) the instantaneous reward gained over the interval $\mathtt{I}$ and specifies it to be in $\mathtt{J}$.[1]

The semantics of CSL formulae are interpreted over the states of the given reward CTMDP $(\mathcal{C}, \mathbf{r})$. Formally, the pair (s, Φ) belongs to the relation $\models_{(\mathcal{C},\mathbf{r})}$, denoted by $s \models_{(\mathcal{C},\mathbf{r})} \Phi$, if and only if Φ is true at s. The index is omitted whenever clear from the context. We need to introduce some additional notation. For state s, let α_s be the Dirac distribution with $\alpha_s(s) = 1$ and 0 otherwise. For a formula Φ, let $Sat(\Phi)$ denote the set of states satisfying Φ, moreover, we let $\mathbf{r}|_{\Phi}$ denote $\mathbf{r}|_{Sat(\Phi)}$. The relation $\models$ is defined as follows:

- Probabilistic Operator: $s \models \mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$ iff for all policies π, it holds:

$$P^{\pi}_{s}(\{\omega \in \Omega \mid \omega \models \Phi\ \mathsf{U}^{\mathtt{I}}\Psi\}) \in \mathtt{J}$$

  where $\omega \models \Phi\ \mathsf{U}^{\mathtt{I}}\Psi$ iff $\exists t \in \mathtt{I}.\ \omega(t) \models \Psi \wedge \forall 0 \leq t' < t.\ \omega(t') \models \Phi$.
- Instantaneous reward: $s \models \mathbb{I}^{t}_{\mathtt{J}}(\Phi)$ iff it holds that $\mathbf{p}^{\pi}_{t} \cdot \mathbf{r}|_{\Phi} \in \mathtt{J}$ for all policies π, where $\mathbf{p}^{\pi}_{t} = \alpha_s \mathbf{V}^{\pi}_{t}$ is the distribution at time t under π, starting with state s.
- Cumulative reward: $s \models \mathbb{C}^{[t,T]}_{\mathtt{J}}(\Phi)$ iff it holds that $(\alpha_s \mathbf{V}^{\pi}_{t}) \cdot \mathbf{g}^{\pi}_{t,T}|_{Sat(\Phi)} \in \mathtt{J}$ for all policies π, where $\mathbf{g}^{\pi}_{t,T}|_{Sat(\Phi)}$ is the gain vector under π as defined in Eqn. (2), with terminal condition $\mathbf{g}_T = 0$.
- Long-run average reward: $s \models \mathbb{S}_{\mathtt{J}}(\Phi)$ iff it holds that $\lim_{T\to\infty} \frac{1}{T} \cdot (\alpha_s \cdot \mathbf{g}^{\pi}_{0,T}|_{\Phi}) \in \mathtt{J}$ for all policies π. This is the average reward gained in an interval with a length going to infinity. In case $\mathbf{r}(s) = 1$ for all $s \in \mathcal{S}$, we refer to $\mathbb{S}$ also as the *steady state probability operator*.

The reward CTMDP satisfies a formula if the initial state does. A few remarks are in order. To simplify the presentation we have skipped the probabilistic next state operator $\mathbb{P}_{\mathtt{J}}(\mathsf{X}^{\mathtt{I}}\,\Phi)$. Recently, the policy classes depending on the whole history, including the complete sequence of visited states, action, sojourn time, has been considered for CTMDPs. This seemingly more powerful class of policies is known to be as powerful as the piecewise constant policies considered in this paper, as shown in [8, 9].

[1] For readers familiar with the PRISM tool notation, $\mathtt{R}_{\mathtt{J}}[\mathtt{C}^{\leq t}]$ corresponds to $\mathbb{C}^{\leq t}_{\mathtt{J}}(true)$, $\mathtt{R}_{\mathtt{J}}[\mathtt{I}^{=t}]$ to $\mathbb{I}^{t}_{\mathtt{J}}(true)$, and $\mathtt{R}_{\mathtt{J}}[\mathtt{S}]$ to $\mathbb{S}_{\mathtt{J}}(true)$, respectively, for CTMCs with rewards.

3.2 Optimal Values and Policies

Our semantics is based on resolving the nondeterministic choices by policies. Obviously, checking probabilistic and reward properties amounts to computing, or approximating, the corresponding optimal values. For the probabilistic operator $\mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$, we define

$$P_s^{max}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) := \sup_{\pi\in\mathcal{M}} P_s^{\pi}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi), \qquad P_s^{min}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) := \inf_{\pi\in\mathcal{M}} P_s^{\pi}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$$

as the *maximal (and minimal) probability* of reaching a Ψ-state along Φ-states. Then, $s \models \mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$ iff $P_s^{max}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) \leq \sup \mathtt{J}$ and $P_s^{min}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) \geq \inf \mathtt{J}$. In case the condition is true, i.e., $\Phi = \mathit{true}$, we refer to it simply as *reachability probability*.

The defined extreme probabilities P_s^{max} and P_s^{min} are also referred to as the *optimal values*. A policy π is called *optimal*, with respect to $\mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$, if it achieves the optimal values, i.e., if $P_s^{\pi}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) = P_s^{max}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$ or $P_s^{\pi}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi) = P_s^{min}(\Phi\ \mathsf{U}^{\mathtt{I}}\Psi)$.

The optimal values and policies are also defined for reward properties in a similar way. Briefly, we define:

- $R_s^{max}(\mathbb{I}^t\,\Phi) = \sup_{\pi\in\mathcal{M}}(\mathbf{p}_t^{\pi}\cdot\mathbf{r}|_{\Phi})$ for instantaneous reward,
- $R_s^{max}(\mathbb{C}^{[t,T]}\,\Phi) = \sup_{\pi\in\mathcal{M}}((\alpha_s\mathbf{V}_t^{\pi})\cdot\mathbf{g}_{t,T}^{\pi}|_{Sat(\Phi)})$ for cumulative reward, and
- $R_s^{max}(\mathbb{S}\,\Phi) = \sup_{\pi\in\mathcal{M}}\left(\lim_{T\to\infty}\frac{1}{T}\left(\alpha_s\cdot\mathbf{g}_{0,T}^{\pi}|_{Sat(\Phi)}\right)\right)$ for long-run average reward.

For the long-run average reward the optimal policy is stationary, which can be computed using a dynamic programming algorithm for average rewards as for example presented in [4]. The optimal policies achieving the supremum (or infimum) for instantaneous and cumulative rewards are piecewise constant, which will become clear in the next section.

4 Model Checking Algorithm

Given a CTMDP $(\mathcal{C},\mathbf{r})$ with reward, a state s, and a CSL formula Φ, the model checking problem asks whether $s \models \Phi$ holds. In this section we present a model checking approach where the basic step consists in characterizing the gain vector for the computation of $R_s^{max}(\mathbb{C}^{\mathtt{I}}\,\Phi)$, $P_s^{max}(\Phi\mathsf{U}^{\mathtt{I}}\Psi)$, and $R_s^{max}(\mathbb{I}^t\,\Phi)$ (Of course, the same holds for the minimal gain vector, which is skipped). The corresponding numerical algorithms shall be presented in the next section.

4.1 Optimal Gain Vector for $R_s^{max}(\mathbb{C}^{\mathtt{I}}\ \mathit{true})$

Our goal is to obtain the vector $\mathbf{g}_0^*$ that corresponds to the maximal gain that can be achieved by choosing an optimal policy in $[0,T]$. Stated differently, for a given $\mathbf{p}_0$, we aim to find a policy π^* which maximizes the gain vector in the interval $[0,T]$ in all elements. It can be shown [12] that this policy is independent of the initial probability vector and we need to find π^* such that

$$\pi^* = \arg\max_{\pi\in\mathcal{M}}\left(\mathbf{V}_T^{\pi}\mathbf{g}_T + \int_0^T \mathbf{V}_t^{\pi}\mathbf{r}\,\mathrm{d}t \quad \text{in all elements}\right). \tag{3}$$

Moreover, the maximal gain vector is denoted by $\mathbf{g}_0^* := \mathbf{g}_0^{\pi^*}$, with $|_G$ omitted as $G = \mathcal{S}$.

The problem of maximizing the accumulated reward of a finite CTMDP in a finite interval $[0, T]$ has been analyzed for a long time. The basic result can be found in [12] and is more than 40 years old. Further results and extensions can be found in [14]. The paper of Miller [12] introduces the computation of a policy π^* which maximizes the accumulated reward in $[0, T]$. The following theorem summarizes the main results of [12], adapted to our setting with a non-zero terminal gain vector $\mathbf{g}_T$:

Theorem 1 (Theorem 1 and 6 of [12]). *Let $(\mathcal{C}, \mathbf{r})$ be a CTMDP with reward, $T > 0$, and let $\mathbf{g}_T$ be the terminal condition of the gain vector. A policy is optimal if it maximizes for almost all $t \in [0, T]$*

$$\max_{\pi \in \mathcal{M}} \left(\mathbf{Q}^{\mathbf{d}_t} \mathbf{g}_t^\pi + \mathbf{r} \right) \text{ where } -\frac{\mathrm{d}}{\mathrm{d}t} \mathbf{g}_t^\pi = \mathbf{Q}^{\mathbf{d}_t} \mathbf{g}_t^\pi + \mathbf{r} \,. \tag{4}$$

There exists a piecewise constant policy $\pi \in \mathcal{M}$ that maximizes the equations.

In [12], the terminal condition $\mathbf{g}_T$ is fixed to the zero vector which is sufficient for the problem considered there. The corresponding proofs can be adapted in a straightforward way for the non-zero $\mathbf{g}_T$. We will see later that a non-zero terminal condition allows us to treat various reachability probabilities as they occur in model checking problems. Recall the vector $\mathbf{g}_t^\pi$ describes the gain at time t, i.e., $\mathbf{g}_t^\pi(s)$ equals the expected reward gained at time T if the CTMDP is in state s at time t and policy π is applied in the interval $[t, T]$. Miller presents a constructive proof of Theorem 1 which defines the following sets for some measurable policy $\pi \in \mathcal{M}$ with gain vector $\mathbf{g}_t^\pi$ at time t.

$$\begin{aligned}
\mathcal{F}_1(\mathbf{g}_t^\pi) &= \left\{ \mathbf{d} \in \mathcal{D} \mid \mathbf{d} \text{ maximizes } \mathbf{q}_{\mathbf{d}}^{(1)} \right\} , \\
\mathcal{F}_2(\mathbf{g}_t^\pi) &= \left\{ \mathbf{d} \in \mathcal{F}_1(\mathbf{g}_t) \mid \mathbf{d} \text{ maximizes } -\mathbf{q}_{\mathbf{d}}^{(2)} \right\} , \\
&\ldots \\
\mathcal{F}_j(\mathbf{g}_t^\pi) &= \left\{ \mathbf{d} \in \mathcal{F}_{j-1}(\mathbf{g}_t) \mid \mathbf{d} \text{ maximizes } (-1)^{j-1} \mathbf{q}_{\mathbf{d}}^{(j)} \right\}
\end{aligned}$$

where

$$\begin{aligned}
&\mathbf{q}_{\mathbf{d}}^{(1)} = \mathbf{Q}^{\mathbf{d}} \mathbf{g}_t^\pi + \mathbf{r} \,, \; \mathbf{q}_{\mathbf{d}}^{(j)} = \mathbf{Q}^{\mathbf{d}} \mathbf{q}^{(j-1)} \text{ and} \\
&\mathbf{q}^{(j-1)} = \mathbf{q}_{\mathbf{d}}^{(j-1)} \text{ for any } \mathbf{d} \in \mathcal{F}_{j-1} (j = 2, 3, \ldots)
\end{aligned}$$

The following theorem results from [12, Lemma 3 and 4].

Theorem 2. *If $\mathbf{d} \in \mathcal{F}_{n+1}(\mathbf{g}_t^\pi)$ then $\mathbf{d} \in \mathcal{F}_{n+k}(\mathbf{g}_t^\pi)$ for all $k > 1$.*

Let π be a measurable policy in $(t', T]$ and assume that $\mathbf{d} \in \mathcal{F}_{n+1}(\mathbf{g}_t^\pi)$ for $t' < t < T$, then exists some ε ($0 < \varepsilon \leq t - t'$) such that $\mathbf{d} \in \mathcal{F}_{n+1}(\mathbf{g}_{t''}^\pi)$ for all $t'' \in [t - \varepsilon, t]$.

We define a *selection procedure* that selects the lexicographically largest vector $\mathbf{d}$ from $\mathcal{F}_{n+1}$ which implies that we define some lexicographical ordering on the vectors $\mathbf{d}$. Then, the algorithm can be defined to get the optimal value with respect to cumulative reward (see [12]), which is presented in Algorithm 1. Let $\mathbf{g}_{t_0}^*$ denote the gain vector at $t = t_0 \geq 0$ and π^* the piecewise constant policy resulting from $\textsc{Optimal}(\mathcal{C}, \mathbf{r}, t_0, T, \mathbf{0})$ of the above algorithm. For the case $t_0 = 0$, the optimal gain for a given initial state s equals then $\alpha_s \mathbf{g}_0^*$. According to the Bellman equations [4] the restriction of the policy

Algorithm 1. OPTIMAL$(\mathcal{C}, \mathbf{r}, t_0, T, \mathbf{g}_T)$: Deciding optimal value and policy

1. Set $t' = T$;
2. Select $\mathbf{d}_{t'}$ using $\mathbf{g}_{t'}$ from $\mathcal{F}_{n+1}(\mathbf{g}_{t'})$ as described ;
3. Obtain $\mathbf{g}_t$ for $0 \leq t \leq t'$ by solving

$$-\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{g}_t = \mathbf{r} + \mathbf{Q}^{\mathbf{d}_{t'}}\mathbf{g}_t$$

   with terminal condition $\mathbf{g}_{t'}$;
4. Set $t'' = \inf\{t : \mathbf{d}_t \text{ satisfies the selection procedure in } (t'', t']\}$;
5. If $t'' > t_0$ go to 2. with $t' = t''$. Otherwise, terminate and return the gain vector $\mathbf{g}^*_{t_0}$ at $t = t_0$ and the resulting piecewise constant policy π^* ;

π^* to the interval $(t, T]$ $(0 < t < T)$ results in an optimal policy with gain vector $\mathbf{g}^*_t$. Observe that Algorithm 1 is not implementable as it is described here, since step 4. cannot be effectively computed. We shall present algorithms to approximate or compute bounds for the optimal gain vector in Section 5.

4.2 Cumulative Reward $R^{max}_s(\mathbb{C}^{\leq t}\, \Phi)$

For computing $R^{max}_s(\mathbb{C}^{\leq t}\, \Phi)$, we have the terminal gain vector $\mathbf{g}_T = \mathbf{0}$. Let $\mathbf{g}^*_0$ denote the gain vector at $t = 0$ and π^* the piecewise constant policy resulting from OPTIMAL$(\mathcal{C}, \mathbf{r}|_\Phi, 0, T, \mathbf{0})$ of the above algorithm. The optimal cumulative reward for a given initial state s equals then $R^{max}_s(\mathbb{C}^{\leq t}\, \Phi) = \alpha_s \mathbf{g}^*_0$.

4.3 Probabilistic Operator $P^{max}_s(\Phi\, \mathsf{U}^{\mathrm{I}} \Psi)$

Let $(\mathcal{C}, \mathbf{0})$ be a CTMDP with zero rewards, $T > 0$. We consider the computation of $P^{max}_s(\Phi\, \mathsf{U}^{\mathrm{I}} \Psi)$, which will be discussed below.

Intervals of the Form $I = [0, T]$. In this case, as for CTMCs [2], once a state satisfying $\neg\Phi \vee \Psi$ has been reached, the future behaviors becomes irrelevant. Thus, these states can be made absorbing by removing all outgoing transitions, without altering the reachability probability. Let $Sat(\Phi)$ denote the set of states satisfying Φ. Applying Theorem 1 for zero-rewards $\mathbf{r} = \mathbf{0}$, with a terminal gain vector $\mathbf{g}_T$, we get directly:

Corollary 1. *Let $\Phi\, \mathsf{U}^{[0,T]} \Psi$ be a CSL path formula with $T > 0$. Let $(\mathcal{C}, \mathbf{0})$ be a CTMDP with zero rewards such that $Sat(\neg\Phi \vee \Psi)$ states are absorbing. Moreover, let $\mathbf{g}_T$ be the terminal gain vector with $\mathbf{g}_T(s) = 1$ if $s \in Sat(\Psi)$ and 0 otherwise. A policy is optimal (w.r.t. $P^{max}_s(\Phi\, \mathsf{U}^{[0,T]} \Psi)$) if it maximizes for almost all $t \in [0, T]$,*

$$\max_{\pi \in \mathcal{M}} \left(\mathbf{Q}^{\mathbf{d}_t}\mathbf{g}^\pi_t\right) \text{ where } -\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{g}^\pi_t = \mathbf{Q}^{\mathbf{d}_t}\mathbf{g}^\pi_t \,. \tag{5}$$

There exists a piecewise constant policy $\pi \in \mathcal{M}$ that maximizes the equations.

The following lemma shows that the optimal gain vector obtained by the above corollary can be used directly to obtain the maximal reachability probability:

Lemma 1. *Let $\mathbf{g}_T$ be the terminal gain vector with $\mathbf{g}_T(s) = 1$ if $s \in Sat(\Psi)$ and 0 otherwise. Assume the procedure* OPTIMAL$(\mathcal{C}, \mathbf{0}, 0, T, \mathbf{g}_T)$ *returns the optimal policy π^* and the corresponding optimal gain vector $\mathbf{g}_0^*$. Then, it holds $P_s^{max}(\Phi\, \mathsf{U}^{[0,T]} \Psi) = \alpha_s \mathbf{g}_0^*$.*

Proof. Since $\mathbf{r} = \mathbf{0}$, Eqn. (3) reduces to $\pi^* = \arg\max_{\pi \in \mathcal{M}} (\mathbf{V}_T^\pi \mathbf{g}_T^\pi$ in all elements). By definition, it is $\mathbf{g}_0^* = \mathbf{V}_T^{\pi^*} \mathbf{g}_T$, which is maximal in all elements. Moreover, since $Sat(\neg\Phi \vee \Psi)$-states are absorbing, the maximal transient probability is the same as the maximal time bounded reachability. Thus, $\mathbf{g}_0^*(s)$ is the maximal probability of reaching $Sat(\Psi)$ within T, along $Sat(\Phi)$-states from s, as $\mathbf{g}_0^*$ is maximal in all elements. Thus, $P_s^{max}(\Phi\, \mathsf{U}^{\mathrm{I}} \Psi) = \alpha_s \mathbf{g}_0^*$. □

Intervals of the form $I = [t_0, T]$ with $t_0 > 0$ and $T \geq t_0$. Let us review the problem of computing an optimal gain vector of a finite CTMDP in a finite interval $[0, T]$ from a new angle. Assume that an optimal policy is known for $[t_0, T]$ and $\mathbf{a}_{[t_0,T]}$ is the optimal gain vector at t_0, then the problem is reduced to finding an extension of the policy in $[0, t_0)$ which means to solve the following maximization problem:

$$\mathbf{g}_0^* = \max_{\pi \in \mathcal{M}} \left(\mathbf{V}_{t_0}^\pi \mathbf{a}_{[t_0,T]} \right). \tag{6}$$

The problem can be easily transferred in the problem of computing the reachability probability for some interval $[t_0, T]$, after a modification of the CTMDP. Essentially, a two step approach has to be taken. As we have seen in Algorithm 1, the optimal policy to maximize the reward is computed in a backwards manner. First the optimal policy is computed for the interval $[t_0, T]$ with respect to the maximal probability $P_s^{max}(\Phi\, \mathsf{U}^{[0,T-t_0]} \Psi)$, using the CTMDP where states from $Sat(\neg\Phi \vee \Psi)$ are made absorbing. This policy defines the vector $\mathbf{a}_{[t_0,T]} = \mathbf{g}_{t_0}$: this is adapted appropriately—by setting the element to 0 for states satisfying $\neg\Phi$—which is then used as terminal condition to extend the optimal policy to $[0, t_0)$ on the original CTMDP.

Let $\mathcal{C}[\Phi]$ denote the CTMDP with states in $Sat(\Phi)$ made absorbing, and let $\mathbf{Q}[\Phi]$ denote the corresponding modified $\mathbf{Q}$-matrix in $\mathcal{C}[\Phi]$. The following corollary summarizes Theorem 1 when it is adopted to the interval bounded reachability probability.

Corollary 2. *Let $(\mathcal{C}, \mathbf{0})$ be a CTMDP with zero rewards $\mathbf{r} = \mathbf{0}$, $t_0 > 0$ and $T \geq t_0$. Let $\Phi\, \mathsf{U}^{[t_0,T]} \Psi$ be a path formula, and $\mathbf{g}_T$ be the terminal gain vector with $\mathbf{g}_T(s) = 1$ if $s \in Sat(\Psi)$ and 0 otherwise. A policy is optimal (w.r.t. $P_s^{max}(\Phi\, \mathsf{U}^{[t_0,T]} \Psi)$) if it*

- *maximizes for almost all $t \in [t_0, T]$*

$$\max_{\pi \in \mathcal{M}} \left(\mathbf{Q}_1^{\mathbf{d}_t} \mathbf{g}_t^\pi \right) \text{ where } -\frac{\mathrm{d}}{\mathrm{d}t} \mathbf{g}_t^\pi = \mathbf{Q}_1^{\mathbf{d}_t} \mathbf{g}_t^\pi \,. \tag{7}$$

  with $\mathbf{Q}_1 := \mathbf{Q}[\neg\Phi \vee \Psi]$ and initial condition at T given by $\mathbf{g}_T$. Note that the vector $\mathbf{g}_{t_0}^*$ is uniquely determined by the above equation.

– *maximizes for almost all* $t \in [0, t_0]$*:*

$$\max_{\pi \in \mathcal{M}} \left(\mathbf{Q}_2^{d_t} \mathbf{g}_t^\pi \right) \text{ where } -\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{g}_t^\pi = \mathbf{Q}_2^{\mathbf{d}_t}\mathbf{g}_t^\pi$$

with $\mathbf{Q}_2 := \mathbf{Q}[\neg\Phi]$*, and initial condition at* t_0 *given by* $\mathbf{g}'$ *defined by:* $\mathbf{g}'(s) = \mathbf{g}^*_{t_0}(s)$ *if* $s \models \Phi$*, and* 0 *otherwise.*

There exists a piecewise constant policy $\pi \in \mathcal{M}$ *that maximizes the equations.*

Notice that the corollary holds for the special case $\Phi = \mathit{true}$ and $t_0 = T$, what we get is also called the *maximal transient probability* of being at $Sat(\Psi)$ at exact time T, namely $\mathbf{V}_T^\pi$ with terminal condition $\mathbf{g}_T$. Now we can achieve the maximal interval bounded reachability probability:

Lemma 2. *Let* $\mathbf{g}_T$ *be as defined in Corollary 2. Assume the procedure* OPTIMAL$(\mathcal{C}[\neg \Phi \vee \Psi], \mathbf{0}, t_0, T, \mathbf{g}_T)$ *returns the optimal policy* $\pi^*_{t_0}$ *and the corresponding optimal gain vector* $\mathbf{g}^*_{t_0}$*. Let* $\mathbf{g}'$ *be defined by* $\mathbf{g}'(s) = \mathbf{g}^*_{t_0}(s)$ *if* $s \models \Phi$*, and* 0 *otherwise.*

Assume the procedure OPTIMAL$(\mathcal{C}[\neg\Phi], \mathbf{0}, 0, t_0, \mathbf{g}')$ *returns the optimal policy* π^* *(extending the policy* $\pi^*_{t_0}$*) and the corresponding optimal gain vector* $\mathbf{g}^*_0$*. Then, it holds* $P_s^{max}(\Phi \,\mathsf{U}^{[t_0,T]} \Psi) = \alpha_s \mathbf{g}^*_0$.

Proof. The optimal gain at time t_0 is obtained by $\mathbf{g}^*_{t_0}$, by Lemma 1. For all $t \leq t_0$, Φ must be satisfied by the semantics for the path formula, thus $\mathbf{g}^*_{t_0}$ is replaced with $\mathbf{g}'$ as initial vector for the following computation. Thus, $\mathbf{g}^*_0 = \mathbf{V}^{\pi^*}_{t_0}\mathbf{g}'$ is maximal in all elements, and $\mathbf{g}^*_0(s)$ is the maximal probability of reaching $Sat(\Psi)$ from s within $[t_0, T]$, along $Sat(\Phi)$ states. Thus, $P_s^{max}(\Phi \,\mathsf{U}^{[t_0,T]} \Psi) = \alpha_s \mathbf{g}^*_0$. □

4.4 Interval Cumulative Reward $R_s^{max}(\mathbb{C}^{\mathtt{I}}\, \Phi)$

The maximal interval cumulative reward $R_s^{max}(C^{\mathtt{I}}\, \Phi)$ can now be handled by combining the cumulative rewards and reachability property. Assume that $I = [t_0, T]$ with $t_0 > 0$ and $T \geq t_0$. As before, we can first compute the cumulative reward between $[t_0, T]$ by $\mathbf{a}_{[t_0,T]} :=$ OPTIMAL$(\mathcal{C}, \mathbf{r}|_\Phi, t_0, T, \mathbf{0})$ (see (6)). So $\mathbf{a}_{[t_0,T]}$ is the maximal cumulative reward between $[t_0, T]$, and the problem now is reduced to finding an extension of the policy in $[0, t_0)$ such that $\mathbf{g}^*_0 = \max_{\pi \in \mathcal{M}} \left(\mathbf{V}^\pi_{t_0} \mathbf{a}_{[t_0,T]} \right)$, which can be seen as reachability probability with terminal condition $\mathbf{a}_{[t_0,T]}$. This value can be computed by OPTIMAL$(\mathcal{C}, \mathbf{0}, 0, t_0, \mathbf{a}_{[t_0,T]})$.

4.5 Instantaneous Reward $R_s^{max}(\mathbb{I}^t\, \Phi)$

Interestingly, the maximal instantaneous reward $\sup_\pi(\mathbf{p}^\pi_t \cdot \mathbf{r}|_\Phi)$ can be obtained directly by OPTIMAL$(\mathcal{C}, \mathbf{0}, 0, t, \mathbf{r}|_\Phi)$. Intuitively, we have a terminal condition given by the reward vector $\mathbf{r}$, and afterwards, it behaves very similar to the same as probabilistic reachability for intervals of the form $[t, t]$.

We have shown that the CSL model checking problem reduces to the procedure OPTIMAL$(\mathcal{C}, \mathbf{r}, t_0, T, \mathbf{g}_T)$. By Theorem 1, an optimal policy existis. Thus, the established connection to the paper by Miller gives another very important implication: namely the existence of finite memory schedulers (each for a nested state subformula) for the CSL formula.

5 Computational Approaches

We now present an improved approach for approximating OPTIMAL such that the error of the final result can be adaptively controlled. It is based on uniformization [16] for CTMCs and its recent extension to CTMDPs with rewards [11], which, in our notation, treats the approximation of $R_s^{max}(\mathbb{C}^{[0,T]}\ true)$.

The optimal policy $\mathbf{g}_t$ and vector are approximated from T backwards to 0 or t_0, starting with some vector $\mathbf{g}_T$ which is known exactly or for which bounds $\underline{\mathbf{g}}_T \leq \mathbf{g}_T \leq \overline{\mathbf{g}}_T$ are known. Observe that for a fixed $\mathbf{d}$ in $(t-\delta, t]$ we can compute $\mathbf{g}_{t-\delta}$ from $\mathbf{g}_t$ as

$$\mathbf{g}^{\mathbf{d}}_{t-\delta} = e^{\delta \mathbf{Q}^{\mathbf{d}}}\mathbf{g}_t + \int_{\tau=0}^{\delta} e^{\tau \mathbf{Q}^{\mathbf{d}}}\mathbf{r}\,\mathrm{d}\tau = \sum_{k=0}^{\infty} \frac{\left(\mathbf{Q}^{\mathbf{d}}\delta\right)^k}{k!}\mathbf{g}_t + \int_{\tau=0}^{\delta}\sum_{k=0}^{\infty} \frac{\left(\mathbf{Q}^{\mathbf{d}}\tau\right)^k}{k!}\mathbf{r}\,\mathrm{d}\tau. \quad (8)$$

We now solve (8) via uniformization [16] and show afterwards how upper and lower bounds for the optimal gain vector can be computed. Let $\alpha_{\mathbf{d}} = \max_{i\in\mathcal{S}}\left(|\mathbf{Q}^{\mathbf{d}}(i,i)|\right)$ and $\alpha = \max_{\mathbf{d}\in\mathcal{D}}(\alpha_{\mathbf{d}})$. Then we can define the following two stochastic matrices for every decision vector $\mathbf{d}$:

$$\mathbf{P}^{\mathbf{d}} = \mathbf{Q}^{\mathbf{d}}/\alpha_{\mathbf{d}} + \mathbf{I} \text{ and } \bar{\mathbf{P}}^{\mathbf{d}} = \mathbf{Q}^{\mathbf{d}}/\alpha + \mathbf{I}. \quad (9)$$

Define the following function to determine the Poisson probabilities in the uniformization approach.

$$\beta(\alpha\delta, k) = e^{-\alpha\delta}\frac{(\alpha\delta)^k}{k!} \text{ and } \zeta(\alpha\delta, K) = \left(1 - \sum_{l=0}^{K}\beta(\alpha\delta, l)\right). \quad (10)$$

Eqns. (9) and (10), combined with the uniformization approach (8), can be used to derive (see [11]) the following sequences of vectors:

$$\mathbf{g}^{\mathbf{d}}_{t-\delta} = \sum_{k=0}^{\infty}\left(\mathbf{P}^{\mathbf{d}}\right)^k\left(\beta(\alpha_{\mathbf{d}}\delta, k)\mathbf{g}_t + \frac{\zeta(\alpha_{\mathbf{d}}\delta, k)}{\alpha_{\mathbf{d}}}\mathbf{r}\right). \quad (11)$$

Assume that bounds $\underline{\mathbf{g}}_t \leq \mathbf{g}^*_t \leq \overline{\mathbf{g}}_t$ are known and define

$$\begin{aligned} &\underline{\mathbf{v}}^{(k)} = \mathbf{P}^{\mathbf{d}_t}\underline{\mathbf{v}}^{(k-1)},\ \underline{\mathbf{w}}^{(k)} = \mathbf{P}^{\mathbf{d}_t}\underline{\mathbf{w}}^{(k-1)} \text{ and} \\ &\overline{\mathbf{v}}^{(k)} = \max_{\mathbf{d}\in\mathcal{D}}\left(\bar{\mathbf{P}}^{\mathbf{d}}\overline{\mathbf{v}}^{(k-1)}\right),\ \overline{\mathbf{w}}^{(k)} = \max_{\mathbf{d}\in\mathcal{D}}\left(\bar{\mathbf{P}}^{\mathbf{d}}\overline{\mathbf{w}}^{(k-1)}\right) \\ &\text{with } \underline{\mathbf{v}}^{(0)} = \underline{\mathbf{g}}_t,\ \overline{\mathbf{v}}^{(0)} = \overline{\mathbf{g}}_t,\ \underline{\mathbf{w}}^{(0)} = \overline{\mathbf{w}}^{(0)} = \mathbf{r}. \end{aligned} \quad (12)$$

If not stated otherwise, we compute $\underline{\mathbf{v}}^k, \underline{\mathbf{w}}^{(k)}$ with $\mathbf{P}^{\mathbf{d}_t}$ where $\mathbf{d}_t$ is the lexicographically smallest vector from $\mathcal{F}_{n+1}(\underline{\mathbf{g}}_t)$. Observe that $\underline{\mathbf{v}}^{(k)}, \underline{\mathbf{w}}^{(k)}$ correspond to a concrete policy that uses decision vector $\mathbf{d}_t$ in the interval $(t-\delta, t]$. Vectors $\overline{\mathbf{v}}^{(k)}, \overline{\mathbf{w}}^{(k)}$ describe some strategy where the decisions depend on the number of transitions which is an ideal case that cannot be improved by any realizable policy. Notice that for zero rewards for probabilistic reachability, we have $\underline{\mathbf{w}}^{(0)} = \overline{\mathbf{w}}^{(0)} = \mathbf{r} = \mathbf{0}$. From the known bounds for $\mathbf{g}^*_t$, new bounds for $\mathbf{g}^*_{t-\delta}$ can then be computed as follows (see [11, Theorem 3]):

$$\begin{aligned}
\underline{\mathbf{g}}^K_{t-\delta} = \sum_{k=0}^{K} \left(\beta(\alpha_{\mathbf{d}}\delta,k)\underline{\mathbf{v}}^{(k)} + \tfrac{\zeta(\alpha_{\mathbf{d}}\delta,k)}{\alpha_{\mathbf{d}}}\underline{\mathbf{w}}^{(k)}\right) + \zeta(\alpha_{\mathbf{d}}\delta,K)\min_{s\in\mathcal{S}}\left(\underline{\mathbf{v}}^{(K)}(s)\right)\mathbb{1} + \\
\left(\delta\zeta(\alpha_{\mathbf{d}}\delta,K) - \tfrac{K+1}{\alpha_{\mathbf{d}}}\zeta(\alpha_{\mathbf{d}}\delta,K+1)\right)\min_{s\in\mathcal{S}}\left(\underline{\mathbf{w}}^{(K)}(s)\right)\mathbb{1} \qquad\qquad \leq \\
\mathbf{g}^*_{t-\delta} \leq \sum_{k=0}^{K} \left(\beta(\alpha\delta,k)\overline{\mathbf{v}}^{(k)} + \tfrac{\zeta(\alpha\delta,k)}{\alpha}\overline{\mathbf{w}}^{(k)}\right) + \zeta(\alpha\delta,K)\max_{s\in\mathcal{S}}\left(\overline{\mathbf{v}}^{(K)}(s)\right) \qquad + \\
\left(\delta\zeta(\alpha\delta,K) - \tfrac{K+1}{\alpha}\zeta(\alpha\delta,K+1)\right)\max_{s\in\mathcal{S}}\left(\overline{\mathbf{w}}^{(K)}(s)\right)\mathbb{1} \quad = \quad \overline{\mathbf{g}}^K_{t-\delta}.
\end{aligned} \tag{13}$$

where $\mathbb{1}$ is a column vector of ones with length n. Before we formulate an algorithm based on the above equation, we analyze the spread of the bounds. If $\underline{\mathbf{g}}_t$ and $\overline{\mathbf{g}}_t$ are upper and lower bounding vectors used for the computation of $\underline{\mathbf{g}}^K_{t-\delta}$ and $\overline{\mathbf{g}}^K_{t-\delta}$, then $\|\overline{\mathbf{g}}_t - \underline{\mathbf{g}}_t\| \leq \|\overline{\mathbf{g}}^K_{t-\delta} - \underline{\mathbf{g}}^K_{t-\delta}\|$ and the additional spread results from the truncation of the Poisson probabilities

$$\begin{aligned}
\varepsilon_{trunc}(t,\delta,K) = \zeta(\alpha\delta,K)\max_{i\in\mathcal{S}}\left(\overline{\mathbf{v}}^{(K)}(i)\right) - \zeta(\alpha_{\mathbf{d}}\delta,K)\min_{i\in\mathcal{S}}\left(\underline{\mathbf{v}}^{(K)}(i)\right) \\
+ \left(\delta\zeta(\alpha\delta,K) - \frac{(K+1)\zeta(\alpha\delta,K+1)}{\alpha}\right)\max_{s\in\mathcal{S}}\left(\overline{\mathbf{w}}^{(K)}(s)\right) \\
- \left(\delta\zeta(\alpha_{\mathbf{d}}\delta,K) - \frac{(K+1)\zeta(\alpha_{\mathbf{d}}\delta,K+1)}{\alpha_{\mathbf{d}}}\right)\min_{s\in\mathcal{S}}\left(\underline{\mathbf{w}}^{(K)}(s)\right)
\end{aligned} \tag{14}$$

and the difference due to the different decisions, denoted by $\varepsilon_{succ}(t,\delta,K) =: \varepsilon^*$, is,

$$\varepsilon^* = \left\|\sum_{k=0}^{K}\left(\beta(\alpha\delta,k)\overline{\mathbf{v}}^{(k)} + \frac{\zeta(\alpha\delta,t)}{\alpha}\overline{\mathbf{w}}^{(k)} - \beta(\alpha_{\mathbf{d}}\delta,k)\underline{\mathbf{v}}^{(k)} - \frac{\zeta(\alpha_{\mathbf{d}}\delta,t)}{\alpha_{\mathbf{d}}}\underline{\mathbf{w}}^{(k)}\right)\right\| \tag{15}$$

where $\mathbf{d}$ is the decision vector chosen by the selection procedure using $\underline{\mathbf{g}}_t$. As shown in [11] the local error of a step of length δ is in $O(\delta^2)$ such that theoretically the global error goes to 0 for $\delta \to 0$. Observe that $\varepsilon_{trunc}(t,\delta,K) \leq \varepsilon_{trunc}(t,\delta,K+1)$, $\varepsilon_{succ}(t,\delta,K) \geq \varepsilon_{succ}(t,\delta,K+1)$ and

$$\begin{aligned}
\varepsilon(t,\delta,K) \quad &= \varepsilon_{trunc}(t,\delta,K) + \varepsilon_{succ}(t,\delta,K) \leq \\
\varepsilon(t,\delta,K+1) &= \varepsilon_{trunc}(t,\delta,K+1) + \varepsilon_{succ}(t,\delta,K+1).
\end{aligned}$$

With these ingredients we can define an adaptive algorithm that computes $\underline{\mathbf{g}}_{t_0}, \overline{\mathbf{g}}_{t_0}$ ($t_0 \leq T$) and a policy π to reach a gain vector of at least $\underline{\mathbf{g}}_{t_0}$ such that $\underline{\mathbf{g}}_{t_0} \leq \mathbf{g}^*_{t_0} \leq \overline{\mathbf{g}}_{t_0}$ and $\|\overline{\mathbf{g}}_{t_0} - \underline{\mathbf{g}}_{t_0}\|_\infty \leq \varepsilon$ for the given accuracy $\varepsilon > 0$.

Algorithm 2 computes bounds for the gain vector with a spread of less than ε, if the time steps become not too small ($< \delta_{\min}$). Parameter ω determines the fraction of the error resulting from truncation of the Poisson probabilities and $K_{\max}$ defines the number of intermediate vectors that are stored. The decision vector for the interval $(t_i, t_{i-1}]$ is stored in $\mathbf{c}_i$. Observe that $t_i < t_{i-1}$ since the computations in the algorithm start at T and end at t_0. The policy defined by the time point t_i and vectors $\mathbf{c}_i$ guarantees a gain vector which is elementwise larger or equal to $\underline{\mathbf{g}}^*_{t_0}$. Parameter $\delta_{\min}$ is used as a lower bound for the time step to avoid numerical underflows. If the Poisson probabilities

Algorithm 2. UNIFORM$(\mathcal{C}, \mathbf{r}, t_0, T, \underline{\mathbf{g}}_T, \overline{\mathbf{g}}_T, \omega, K_{\max}, \varepsilon)$: Bounding vectors for $\mathbf{g}^*_{t_0}$

1. initialize $i = 0$ and $t = T$;
2. set $stop = false$, $K = 1$ and $\underline{\mathbf{v}}^{(0)} = \underline{\mathbf{g}}_t$, $\overline{\mathbf{v}}^{(0)} = \overline{\mathbf{g}}_t$, $\underline{\mathbf{w}}^{(0)} = \overline{\mathbf{w}}^{(0)} = \mathbf{r}$;
3. select $\mathbf{d}_t$ from $\mathcal{F}_{n+1}(\mathbf{g}_t)$ as described and if $i = 0$ let $\mathbf{c}_0 = \mathbf{d}_t$;
4. repeat
5. compute $\underline{\mathbf{v}}^{(K)}, \overline{\mathbf{v}}^{(K)}, \underline{\mathbf{w}}^{(K)}, \overline{\mathbf{w}}^{(K)}$ using (12);
6. find $\delta = \max\left(\arg\max_{\delta' \in [0,t]} \left(\varepsilon_{trunc}(t, \delta', K) \leq \frac{\omega \delta'}{T - t_0}\varepsilon\right), \min\left(\delta_{\min}, t - t_0\right)\right)$;
7. compute $\varepsilon_{trunc}(t, \delta, K)$ and $\varepsilon_{succ}(t, \delta, K)$ using (14,15) ;
8. if $\varepsilon_{trunc}(t, \delta, K) + \varepsilon_{succ}(t, \delta, K) > \frac{T - t + \delta}{T - t_0}\varepsilon$ then
9. reduce δ until
   $\varepsilon_{trunc}(t, \delta, K) + \varepsilon_{succ}(t, \delta, K) \leq \frac{T - t + \delta}{T - t_0}\varepsilon$
   or $\delta = \min\left(\delta_{\min}, t - t_0\right)$ and set $stop = true$;
10. else
11. $K = K + 1$;
12. until $stop$ or $K = K_{\max} + 1$;
13. compute $\underline{\mathbf{g}}_{t-\delta}$ from $\underline{\mathbf{v}}^{(k)}, \underline{\mathbf{w}}^{(k)}$ and $\overline{\mathbf{g}}_{t-\delta}$ from $\overline{\mathbf{v}}^{(k)}, \overline{\mathbf{w}}^{(k)}$ $(k = 0, \ldots, K)$ using (13);
14. if $\mathbf{d}_t \neq \mathbf{c}_i$ then $\mathbf{c}_{i+1} = \mathbf{d}_t$, $t_i = t - \delta$ and $i = i + 1$;
15. if $t - t_0 = \delta$ then terminate else go to 2. with $t = t - \delta$;

are computed with the algorithm from [17], then all computations are numerically stable and use only positive values. A non-adaptive version of the algorithm can be realized by fixing the number of iterations used in the loop between 4. and 12.

To verify a property that requires a reward to be smaller than some threshold value, the computed upper bound has to be smaller than the threshold. If the lower bound is larger than the required value, then the property is disproved, if the threshold lies between lower and upper bound, no decision about the property is possible.

6 Case Studies

We implemented our model checking algorithm in an extension of the probabilistic model checker MRMC [18]. In addition, we implemented a method to compute long-run average state probabilities [3]. The implementation is written in C, using sparse matrices. Parallelism is not exploited. All experiments are performed on an Intel Core 2 Duo P9600 with 2.66 GHz and 4 GB of RAM running on Linux.

6.1 Introductory Example

We consider a simple example taken from [19], which is shown in Figure 1. We consider a single atomic proposition s_4 which holds only in state s_4.

First we analyze the property $\mathbb{P}_{<x}(\Diamond^{[0,T]} s_4)$ for state s_1. In this case, state s_4 is made absorbing by removing the transition from s_4 to s_1 (shown as a dashed line in the figure), as discussed in Subs. 4.3. Table 1 contains the results and efforts to compute the maximal reachability probabilities for $T = 4$ and 7 with the adaptive and non-adaptive variant of the uniformization approach. The time usage is given in seconds. It can be seen that the adaptive version is much more efficient and should be the method of choice in this example. The value of ε that is required to prove $\mathbb{P}_{<x}(\Diamond^{[0,T]} s_4)$ depends on x. E.g., if $T = 4$ and $x = 0.672$, then $\varepsilon = 10^{-4}$ is sufficient whereas $\varepsilon = 10^{-3}$ would not allow one to prove or disprove the property.

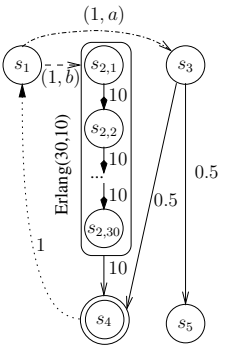

Fig. 1. A CTMDP

Table 1. Bounds for the probability of reaching s_4 in $[0, T]$, i.e., $P^{max}_{s_1}(\Diamond^{[0,T]} s_4)$

		Uniformization $K = 5$					Uniformization $K_{\max} = 20, \omega = 0.1$				
T	ε	lower	upper	steps	iter	time	lower	upper	steps	iter	time
4.0	10^{-4}	0.671701	0.671801	720	3600	0.03	0.671772	0.671803	211	774	0.02
4.0	10^{-5}	0.671771	0.671781	5921	29605	0.10	0.671778	0.671781	2002	5038	0.09
4.0	10^{-6}	0.671778	0.671779	56361	281805	0.87	0.671778	0.671779	19473	40131	0.63
7.0	10^{-4}	0.982746	0.982846	1283	6415	0.04	0.982836	0.982846	364	1333	0.04
7.0	10^{-5}	0.982835	0.982845	10350	51750	0.22	0.982844	0.982845	3463	8098	0.19
7.0	10^{-6}	0.982844	0.982845	97268	486340	1.64	0.982845	0.982845	33747	68876	1.50

Table 2. Bounds for reaching s_4 in $[3, 7]$, i.e., $P^{max}_{s_1}(\Diamond^{[t_0,T]} s_4)$

	$\varepsilon = 1.0e-3$				$\varepsilon = 6.0e-4$			
ε_1	time bounded prob.		$iter_1$	$iter_2$	time bounded prob.		$iter_1$	$iter_2$
$9.0e-4$	0.97170	0.97186	207	90	–	–	–	–
$5.0e-4$	0.97172	0.97186	270	89	0.97176	0.97185	270	93
$1.0e-4$	0.97175	0.97185	774	88	0.97178	0.97185	774	91
$1.0e-5$	0.97175	0.97185	5038	88	0.97179	0.97185	5038	91

To compute the result for $\mathbb{P}_{<x}(\Diamond^{[t_0,T]} s_4)$, the two step approach is used. We consider the interval $[3, 7]$. Thus, in a first step the vector $\mathbf{a}_{[3,7]}$ is computed from the CTMDP where s_4 is made absorbing. Then the resulting vectors $\underline{\mathbf{g}}_3$ and $\overline{\mathbf{g}}_3$ are used as terminal conditions to compute $\underline{\mathbf{g}}_0$ and $\overline{\mathbf{g}}_0$ from the original process including the transition between s_4 and s_1. Apart from the final error bound ε for the spread between $\underline{\mathbf{g}}_0$ and $\overline{\mathbf{g}}_0$, an additional error bound ε_1 ($< \varepsilon$) has to be defined which defines the spread between $\underline{\mathbf{g}}_3$ and $\overline{\mathbf{g}}_3$. Table 2 includes some results for different values of ε and ε_1. The column headed with $iter_i$ ($i = 1, 2$) contains the number of iterations of the i-th phase. It can be seen that for this example, the first phase requires more effort such that ε_1 should be chosen only slightly smaller than ε to reduce the overall number of iterations.

Here, it is important to take time-dependent policies to arrive at truly maximal reachability probabilities. The maximal value obtainable for time-abstract policies (using a recent algorithm for CTMDPs [6, 18]) are 0.584284 (versus 0.6717787) for a time bound of 4.0, and 0.9784889 (versus 0.9828449) for a time bound of 7.0.

6.2 Work Station Cluster

As a more complex example, we consider a fault-tolerant workstation cluster (FTWC), in the form considered in [18]. Time bounded reachability analysis for this model was thus far restricted to time-abstract policies [18], using a dediated algorithm for uniform CTMDPs [5]. In a uniform CTMDP (including the one studied here) rate sums are identical across states and nondeterministic choices, and this can be exploited in the algorithm. The general design of the workstation cluster is shown in Fig. 2. It consists of two sub-clusters which are connected via a backbone. There are N workstations in each sub-cluster which are connected together in a star-topology with a switch as central node. The switches provide additionally the interface to the backbone. Each of the components in the fault-tolerant workstation cluster can break down (fail) with a given rate and then needs to be repaired before becoming available again. There is a single repair unit for the entire cluster, not depicted in the figure, which is only capable of repairing one failed component at a time, with a rate depending on the component. When multiple components are down, there is a non-deterministic decision to be taken which of the failed components is to be repaired next.

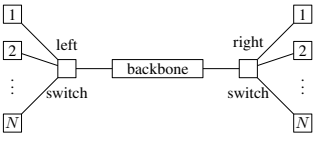

Fig. 2. FTWC structure

We say that our system provides *premium* service whenever at least N workstations are operational. These workstations have to be connected to each other via operational switches. When the number of operational workstations in one sub-cluster is below N, premium quality can be ensured by an operational backbone under the condition that there are at least N operational workstations in total. We consider these properties:

$P1$: Probability to reach non-premium service within time T: $\mathbb{P}_{<x}(\Diamond^{[0,T]}\neg premium)$,
$P2$: Steady-state probability of having non-premium service: $\mathbb{S}_{<x}(\neg premium)$,
$P3$: Steady-state probability of being in a state where the probability to reach non-premium service within time T is above $\frac{1}{2}$: $\mathbb{S}_{<x}(\neg\mathbb{P}_{<\frac{1}{2}}(\Diamond^{[0,T]}\neg premium))$.

Results and statistics are reported in Table 3. For $P1$, we also give numbers for time-abstract policy-based computation exploiting model uniformity [5]. We chose $\varepsilon = 10^{-6}$ and $K_{\max} = 70$. As we see, for $P1$ the probabilities obtained using time-abstract and general policies agree up to ε, thus time-abstract policies seem sufficient to obtain maximal reachability probabilities for this model and property, opposed to the previous example. Our runtime requirements are higher than what is needed for the time-abstract policy class, if exploiting uniformity [5]. Without uniformity exploitation [6], the time-abstract computation times are worse by a factor of 100 to 100,000 compared to our analysis (yielding the same probability result, not shown in the table). However, even for the largest models and time bounds considered, we were able to obtain precise

Table 3. Statistics for the FTWC analysis. For $N = 16$, $N = 64$ and $N = 128$, the state space cardinality is 10130, 151058 and 597010, respectively.

		P1		P1 time-abstract		P2	P3	
$\downarrow N$	$T\rightarrow$	500h	5000h	500h	5000h		500h	5000h
16	time	1s	9s	0s	1s	0s	1s	9s
	prob.	0.0381333	0.3243483	0.0381323	0.3243474	0.0003483	0.0003483	0.0003526
64	time	21s	3m 28s	3s	7s	14s	33s	3m 31s
	prob.	0.1228243	0.7324406	0.1228233	0.7324401	0.0012808	0.0018187	1.0
128	time	2m 46s	34m 5s	13s	40s	1m 30s	4m 8s	35m 9s
	prob.	0.1837946	0.8698472	0.1837937	0.8698468	0.0020517	0.0037645	1.0

results within reasonable time, which shows the practical applicability of the method. Long-run properties $P2$ and nested variation $P3$ can be handled in a similar amount of time, compared to $P1$.

6.3 Further Empirical Evaluation

Further empirical evaluations can be found at

`http://depend.cs.uni-saarland.de/tools/ctmdp`.

The results are generally consistent with the above experiments. As an example, Table 4 lists some runtimes for the European Train Control System (ETCS) case [20]. Details for the model can be found on the website. The property considered is $P_{<x}(\Diamond^{[0,T]}\ unsafe)$, corresponding to the maximal probability that a train must break within T hours of operation. The model consists of "#tr." trains, that are affected by failures. Failure delay distributions are given by Erlang distributions with "#ph." phases. As can be seen, the algorithm for time dependent scheduler analysis is slower than the simpler time-independent analysis, but scales rather smoothly.

Table 4. ETCS Runtimes

			time-dep.		time-abs.	
#tr.	#ph.	#states	10h	180h	10h	180h
3	5	21722	5s	1m 22s	2s	22s
3	10	56452	14s	3m 41s	4s	1m 1s
4	5	15477	4s	59s	1s	16s
4	10	59452	15s	4m 2s	5s	1m 8s

7 Related Work

Our paper builds on the seminal paper of Miller [12]: the problem studied there can be considered as the reward operator $\mathbb{C}^{[0,T]}_{[0,x]}(true)$. Time-bounded reachability for CTMDPs in the context of model checking has been studied, restricted to uniform CTMDPs and for a restricted, time-abstract, class of policies [5]. These results have later been extended to non-uniform stochastic games [6]. Time-abstract policies are strictly less powerful than time-dependent ones [5], considered here and in [7].

Our logic is rooted in [15]. Restricting to CTMCs with or without rewards, the semantics coincides with the standard CSL semantics, as in [15]. However, it is interesting to note that our semantics is defined without refering to *timed paths*, in contrast to

established work (e.g. [2]). This twist enables a drastically simplified presentation. The logic in [15] has a more general probabilistic operator of the form $\mathbb{P}_{\mathtt{J}}(\Phi\ \mathsf{U}^{\mathtt{I}}_{\mathtt{K}}\Psi)$ which allows one to constrain the reward accumulated prior to satisfying Ψ to lie in the interval $\mathtt{K}$. Our framework can not be applied directly to those properties, which we consider as interesting future work.

So far, the common approach to obtain the optimal gain vector proceeds via an approximate *discretization* using a fixed interval of length h, instead of computing t'' as in Algorithm 1. As shown in [12] and also for a slightly different problem in [21], this approach converges towards the optimal solution for $h \to 0$. Let λ be the maximal exit rate in matrix $\mathbf{Q}^{\mathbf{d}}$ for some decision vector $\mathbf{d}$. For probabilistic reachability with interval $[0, T]$, namely $\mathbb{P}^{max}_{s}(\Diamond^{[0,T]}\Phi)$, the number of steps is shown to be bounded by $\mathcal{O}((\lambda T)^2/\varepsilon)$ in [9], to guarantee global accuracy ε. Recently, this bound was further improved to $\mathcal{O}(\lambda T/\varepsilon)$ [10].

The approach presented here is much more efficient than the discretization technique in [9, 10]. As an example we reconsider our introductory example. Discretization requires $iter \approx \lambda T/\varepsilon$ iterations to reach a global accuracy of ε. For $\lambda = 10$, $T = 4$ and $\varepsilon = 0.001$, uniformization requires 201 iterations whereas the discretization approach would need about $40,000$ iterations. For $T = 7$ and for $\varepsilon = 10^{-6}$, uniformization needs $68,876$ iterations, whereas discretization requires about $70,000,000$ iterations to arrive at comparable accuracy, thus the difference is a factor of 1000.

8 Conclusions

The paper presents a new approach to model checking CSL formulae over CTMDPs. A computational approach based on uniformization enables the computation of time bounded reachability probabilities and rewards accumulated during some finite interval. It is shown how these values can be used to prove or disprove CSL formulae. The proposed uniformization technique allows one to compute results with a predefined accuracy that can be chosen with respect to the CSL formula that has to be proved. The improvements resemble the milestones in approximate CTMC model checking research, which was initially resorting to discretization [13], but got effective only through the use of uniformization [2].

The uniformization algorithm approximates, apart from the bounds for the gain vector, also a policy that reaches the lower bound gain vector. This policy is not needed for model checking a CSL formula but it is, of course, of practical interest since it describes a control strategy which enables a system to obtain the required gain—up to ε.

Finally, we note that the current contribution of our paper can be combined with three-value CSL model checking by Katoen *et al* [22], to attenuate the well-known robustness problem of nested formulae in stochastic model checking. For the inner probabilistic state formulae, our algorithm will compute the corresponding probability—up to ε. Using the method in [22] we obtain a three-valued answer, either yes/no, or "don't-know". Then, if we come to the outermost probabilistic operator, we will compute an upper and lower bound of the probabilities. We get a three-valued answer again. In case of a don't-know answer for a state we want to check, we can reduce ε to decrease the number of don't-know states for the inner probabilistic formulae.

Acknowledgement. Ernst Moritz Hahn and Holger Hermanns are partially supported by the DFG/NWO Bilateral Research Programme ROCKS, by the DFG as part of SFB/TR 14 AVACS, and by the EC FP-7 programme under grant agreement no. 214755 - QUASIMODO. Lijun Zhang is partially supported by MT-LAB, a VKR Centre of Excellence.

References

1. Aziz, A., Sanwal, K., Singhal, V., Brayton, R.K.: Model-checking continuous-time Markov chains. ACM Trans. Comput. Log. 1, 162–170 (2000)
2. Baier, C., Haverkort, B.R., Hermanns, H., Katoen, J.P.: Model-checking algorithms for continuous-time Markov chains. IEEE Trans. Software Eng. 29, 524–541 (2003)
3. Howard, R.A.: Dynamic Programming and Markov Processes. John Wiley and Sons, Inc., Chichester (1960)
4. Bertsekas, D.P.: Dynamic Programming and Optimal Control. Athena Scientific, Belmont (2005)
5. Baier, C., Hermanns, H., Katoen, J.P., Haverkort, B.R.: Efficient computation of time-bounded reachability probabilities in uniform continuous-time Markov decision processes. Theor. Comput. Sci. 345, 2–26 (2005)
6. Brázdil, T., Forejt, V., Krcal, J., Kretínský, J., Kucera, A.: Continuous-time stochastic games with time-bounded reachability. In: FSTTCS. LIPIcs, vol. 4, pp. 61–72 (2009)
7. Neuhäußer, M.R., Stoelinga, M., Katoen, J.-P.: Delayed nondeterminism in continuous-time markov decision processes. In: de Alfaro, L. (ed.) FOSSACS 2009. LNCS, vol. 5504, pp. 364–379. Springer, Heidelberg (2009)
8. Rabe, M., Schewe, S.: Finite optimal control for time-bounded reachability in CTMDPs and continuous-time Markov games. In: CoRR, pp. 1004–4005 (2010)
9. Neuhäußer, M.R., Zhang, L.: Time-bounded reachability probabilities in continuous-time Markov decision processes. In: QEST (2010)
10. Chen, T., Han, T., Katoen, J.P., Mereacre, A.: Computing maximum reachability probabilities in Markovian timed automata. Technical report, RWTH Aachen (2010)
11. Buchholz, P., Schulz, I.: Numerical analysis of continuous time Markov decision processes over finite horizons. Computers & Operations Research 38, 651–659 (2011)
12. Miller, B.L.: Finite state continuous time Markov decision processes with a finite planning horizon. SIAM Journal on Control 6, 266–280 (1968)
13. Baier, C., Katoen, J.-P., Hermanns, H.: Approximate symbolic model checking of continuous-time markov chains (Extended abstract). In: Baeten, J.C.M., Mauw, S. (eds.) CONCUR 1999. LNCS, vol. 1664, p. 146. Springer, Heidelberg (1999)
14. Lembersky, M.R.: On maximal rewards and ε-optimal policies in continuous time Markov decision chains. The Annals of Statistics 2, 159–169 (1974)
15. Baier, C., Haverkort, B.R., Hermanns, H., Katoen, J.-P.: On the logical characterisation of performability properties. In: Welzl, E., Montanari, U., Rolim, J.D.P. (eds.) ICALP 2000. LNCS, vol. 1853, p. 780. Springer, Heidelberg (2000)
16. Gross, D., Miller, D.: The randomization technique as a modeling tool and solution procedure for transient Markov processes. Operations Research 32, 926–944 (1984)
17. Fox, B.L., Glynn, P.W.: Computing Poisson probabilities. Comm. ACM 31, 440–445 (1988)
18. Katoen, J.P., Zapreev, I.S., Hahn, E.M., Hermanns, H., Jansen, D.N.: The ins and outs of the probabilistic model checker MRMC. In: QEST, pp. 167–176 (2009)

19. Zhang, L., Neuhäußer, M.R.: Model checking interactive markov chains. In: Esparza, J., Majumdar, R. (eds.) TACAS 2010. LNCS, vol. 6015, pp. 53–68. Springer, Heidelberg (2010)
20. Böde, E., et al.: Compositional performability evaluation for statemate. In: QEST, pp. 167–178 (2006)
21. Martin-Löfs, A.: Optimal control of a continuous-time Markov chain with periodic transition probabilities. Operations Research 15, 872–881 (1967)
22. Katoen, J.-P., Klink, D., Leucker, M., Wolf, V.: Three-valued abstraction for continuous-time markov chains. In: Damm, W., Hermanns, H. (eds.) CAV 2007. LNCS, vol. 4590, pp. 311–324. Springer, Heidelberg (2007)