Observing Continuous-Time MDPs by 1-Clock Timed Automata*

Taolue Chen[1], Tingting Han[1], Joost-Pieter Katoen[2], and Alexandru Mereacre[1]

[1] Department of Computer Science, University of Oxford, United Kingdom
[2] Software Modeling and Verification Group, RWTH Aachen University, Germany

Abstract. This paper considers the verification of continuous-time Markov decision process (CTMDPs) against single-clock deterministic timed automata (DTA) specifications. The central issue is to compute the maximum probability of the set of timed paths of a CTMDP $\mathcal{C}$ that are accepted by a DTA $\mathcal{A}$. We show that this problem can be reduced to a linear programming problem whose coefficients are maximum timed reachability probabilities in a set of CTMDPs, which are obtained via a graph decomposition of the product of the CTMDP $\mathcal{C}$ and the region graph of the DTA $\mathcal{A}$.

1 Introduction

Markov decision processes (MDPs) are a prominent mathematical system model for modeling decision-making—modeled as nondeterministic choices—in situations where outcomes are partly random and partly under the control of a decision maker [24]. MDPs, also referred to as turn-based $1\frac{1}{2}$-player games, are intensively used in decision making and planning with a focus on optimization problems which are typically solved via dynamic programming. They are a discrete-time stochastic control process where at each time step, the decision maker (i.e., the scheduler) may select any action α that is enabled in the current state s. The MDP reacts on this choice by probabilistically moving to state s' with probability $\mathbf{P}(s, \alpha, s')$. A discrete-time Markov chain (DTMC) is an MDP where for each state only a single action is enabled. Since the mid-eighties, MDPs (and DTMCs as special subclass) have been the active subject of applying model checking. Whereas the initial focus was on qualitative properties (e.g., "can a state be reached almost surely, i.e., with probability one?"), the emphasis soon shifted towards *quantitative* properties. Several specification formalisms have been adopted, such as LTL [34,19], probabilistic versions of CTL [9,6], as well as automata [19,21]. The key issue in the quantitative verification of MDPs is to determine the maximum, or dually, minimum probability of a certain event of interest, such as $\Diamond G$, $\Box\Diamond G$, and so forth, where G is a set of states which is either given explicitly or as a state formula. For finite-state MDPs, it is well-known that e.g., extremum reachability probabilities can be obtained by solving linear

* This research is partially supported by the EU FP7 Project MoVeS and the ERC Advanced Grant VERIWARE.

G. Delzanno and I. Potapov (Eds.): RP 2011, LNCS 6945, pp. 2–25, 2011.

programming (LP) problem and that memoryless schedulers suffice to obtain such extrema. If the reachability event is constrained by the maximum number of allowed transitions, one has to resort to finite-memory schedulers, but still a simple value iteration technique suffices to compute the extremum probabilities with the required accuracy. Such techniques have been implemented in model checkers such as PRISM[1] and LIQUOR [18] and successfully applied to several practical case studies such as randomized distributed protocols.

Continuous-time Markov decision processes (CTMDPs) [32] extend MDPs by associating a random delay in state s on selecting action α by the scheduler. Choosing action α in state s yields a random delay in s by the CTMDP which is governed according to an exponential distribution with rate $r^{\alpha}(s)$. Thus, the probability to wait at most d time units in state s on choosing α is $1 - e^{-r^{\alpha}(s) \cdot d}$. After delaying, a CTMDP evolves like an MDP probabilistically to state s' with probability $\mathbf{P}(s, \alpha, s')$. A continuous-time Markov chain (CTMC) is a CTMDP where for each state only a single action is enabled. The state residence time in a CTMC is thus independent of the action chosen. CTMCs have received quite some attention by the verification community since the late nineties. This work has primarily focused on CSL (Continuous Stochastic Logic), a timed probabilistic version of the branching-time temporal logic CTL. The key issue in CSL model checking is to compute the probability of the event $\Diamond^{\leq T} G$ where $T \in \mathbb{R}_{\geq 0}$ acts as a time bound. It has been shown that such probabilities can be characterized as least solution of Volterra integral equation systems and can be computed in a numerically stable and efficient way by reducing the problem to transient analysis of CTMCs [4]. This has been implemented in model checkers such as MRMC [25][2] and PRISM, and has been applied successfully to several cases from systems biology and queueing theory, to mention a few.

Recently, the verification of CTMCs has been enriched by considering linear-time properties equipped with timing constraints. In particular, [15,16] treat linear real-time specifications that are given as *deterministic timed automata* (DTA) [2]. DTA are automata equipped with clock variables that can be used to measure the elapse of time, can be reset to zero, and whose value can be inspected in transition guards. The fact that these automata are deterministic means that for any clock valuation and state, the successor state is uniquely determined. Whereas timed automata are typically used as system models describing the possible system behaviors, we use them—in analogy to [1]—as objectives that need to be fulfilled by the system. In our context, DTA specifications include properties of the form "what is the probability to reach a given target state within the deadline, while avoiding unsafe states and not staying too long in any of the dangerous states on the way?". DTA have recently also been adopted as specification language for generalized semi-Markov processes (and their game extensions) in [11,12]. The central issue in checking a DTA specification is computing the probability of the set of paths in a CTMC that are accepted by the DTA. This can be reduced to computing the (simple) reachability probability

[1] `http://www.prismmodelchecker.org/`
[2] `http://www.mrmc-tool.org/trac/`

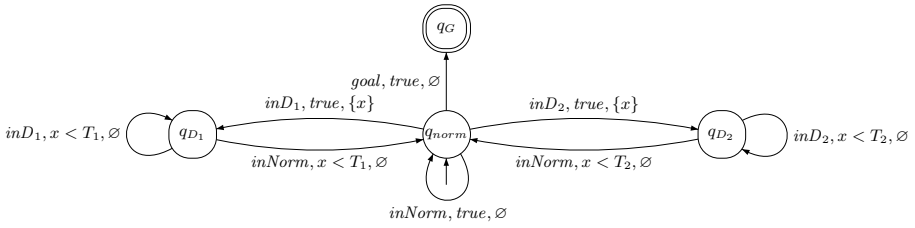


Fig. 1. An example 1-clock DTA that goes beyond timed reachability

in a (somewhat simplified variant of) *piecewise deterministic Markov process* (PDP, [20]), basically a stochastic hybrid model which is obtained by a synchronous product construction between the CTMC and the region graph of the DTA [16]. A prototypical implementation of this technique has recently been presented [7] and has led to the efficient verification of CTMCs of several hundreds of thousands of states against one-clock DTA specifications. The appealing properties of this algorithm are that it resorts to standard computational procedures, i.e., graph analysis, region graph construction, solving systems of linear equations, and transient analysis of CTMCs for which efficient algorithms exist.

In contrast to MDPs, CTMDPs have received far less attention by the verification community; in fact, the presence of nondeterminism and continuous time makes their analysis non-trivial. CTMDPs have originated as continuous-time variants of finite-state probabilistic automata [26], and have been used for, among others, the control of queueing systems, epidemic, and manufacturing processes. Their analysis is mainly focused on determining optimal schedulers for criteria such as expected total reward and expected (long-run) average reward, cf. the survey [23]. The formal verification of CTMDPs has mostly concentrated on computing extremum probabilities for the event $\Diamond^{\leq T} G$ with time bound $T \in \mathbb{R}_{\geq 0}$. Whereas memoryless schedulers suffice for extremum reachability probabilities in MDPs, maximizing (or minimizing) timed reachability probabilities requires *timed* schedulers, i.e., schedulers that "know" how much time has elapsed so far [29,28,8]. As these schedulers are infinite objects, most work has concentrated on obtaining ϵ-optimal schedulers—mostly piecewise-constant schedulers that only change finitely often in the interval $[0, T]$—that approximate the extremum probability obtained by a timed scheduler up to a given accuracy $\epsilon > 0$ [31,33]. Recently, the use of adaptive uniformization has been proposed as an alternative numerical approach to obtain such ϵ-optimal schedulers [13]. Another approach is to concentrate on sub-optimal schedulers, and consider the optimal *time-abstract* scheduler [5,10]. This is a much simpler and efficient procedure that does not rely on discretization, and in several cases suffices. Some of the techniques for both timed and time-abstract schedulers have recently been added to the model checker MRMC [25].

In this paper, we concentrate on a larger class of properties and consider the verification of CTMDPs against linear real-time specifications given as *single-clock* DTA. Note that single-clock DTA cover a whole range of safety and liveness

objectives and naturally include timed reachability objectives such as $\Diamond^{\leq T} G$. We believe that DTA are a very natural specification formalism that captures a rich set of practically interesting properties. For instance, Fig. 1 presents an example 1-clock DTA that goes beyond timed reachability properties. It asserts "reach a given target G (modeled by state q_G) while not staying too long (at most T_1 and T_2 time units in respective zones D_1 and D_2) in any of the two "dangerous zones on the way". For simplicity, we assume the dangerous zones D_1 and D_2 are not adjacent. In case the system stays too long in one of the dangerous zones, it resides in either location q_{D_1} or q_{D_2} forever, and will never reach the goal state. This property can neither be expressed in CSL nor in one of its existing dialects [3,22]. The central issue now in checking such a DTA specification is computing the extremum probability of the set of paths in a CTMDP $\mathcal{C}$ that are accepted by the DTA $\mathcal{A}$. We show that the approach in [15,16,7] can be adapted to this problem in the following way. We first establish that the extremum probability of CTMDP $\mathcal{C}$ satisfying DTA $\mathcal{A}$ can be characterized as the extremum reachability probability in the product of $\mathcal{C}$ and the region graph of $\mathcal{A}$. Here, the region graph is based on a variant of the standard region construction for timed automata [2]. The product $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$ is in fact a simple instance of a piecewise deterministic Markov *decision* process (PDDP, [20]). The extremum reachability probabilities in $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$ are then characterized by a Bellman equation. These results so far are also applicable to DTA with an arbitrary number of clocks (although formulated in this paper for single-clock DTA only). For 1-clock DTA, we then show that solving this Bellman equation can be reduced to an LP problem whose coefficients are extremum timed reachability probabilities in the CTMDP $\mathcal{C}$, i.e., events of the form $\Diamond^{\leq T} G$. The size of the obtained LP problem is in $\mathcal{O}(|S|\cdot|Q|\cdot m)$, where S is the state space of CTMDP $\mathcal{C}$, Q is the state space of DTA $\mathcal{A}$, and m is the number of distinct constants appearing in the guards of $\mathcal{A}$.

To put in a nutshell, this paper shows that the verification of CTMDPs against 1-clock DTA objectives can be done by a region graph construction, a product construction, and finally solving an LP problem whose coefficients are extremum timed reachability probabilities in CTMDPs. 1-clock DTA objectives model a rich class of interesting properties in a natural manner and include timed reachability. To the best of our knowledge, this is the first work towards treating linear real-time objectives of CTMDPs. The main appealing implication of our result is that CTMDPs can be verified against 1-clock DTA objectives using rather standard means. The availability of the first practical implementations for timed reachability of CTMDPs paves the way to a realization of our approach in a prototypical tool.

Organization of this paper. Section 2 defines the basic concepts for this paper: CTMDPs, DTA, and formalizes the problem tackled in this paper. Section 3 shortly recapitulates a mathematical characterization of maximum timed reachability probabilities in CTMDPs. Section 4 introduces the product $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$ and provides a Bellman equation for reachability events in this product. Section 5 is the core of this paper, and shows that for 1-clock DTA, the solution of the Bellman equation can be obtained by solving an LP problem whose

coefficients are extremum timed reachability probabilities in CTMDPs obtained from $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$. Section 6 concludes the paper. The proof of Theorem 2 is included in the appendix.

2 Preliminaries

Given a set H, let $\Pr : \mathcal{F}(H) \to [0,1]$ be a probability measure on the measurable space $(H, \mathcal{F}(H))$, where $\mathcal{F}(H)$ is a σ-algebra over H.

2.1 CTMDP

Let AP be a fixed, finite set of atomic propositions.

Definition 1 (CTMDP). *A* continuous-time Markov decision process *is a tuple* $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$*, where*

- S *is a finite set of* states*;*
- s_0 *is the* initial state*;*
- Act *is a finite set of* actions*;*
- $\mathbf{P} : S \times Act \times S \to [0,1]$ *is a* transition probability matrix*, such that for any state* $s \in S$ *and action* $\alpha \in Act$*,* $\sum_{s' \in S} \mathbf{P}(s, \alpha, s') \in \{0, 1\}$*;*
- $r : S \times Act \to \mathbb{R}_{\geq 0}$ *is an* exit rate function*; and*
- $L : S \to 2^{\mathrm{AP}}$ *is a* labeling function*.*

The set of actions that are enabled in state s is denoted $Act(s) = \{\, \alpha \in Act \mid r^\alpha(s) > 0 \,\}$ where $r^\alpha(s)$ is a shorthand for $r(s, \alpha)$. The operational behavior of a CTMDP is as follows. On entering state s, an action α, say, in $Act(s)$ is non-deterministically selected. The CTMDP now evolves probabilistically as follows. Given that action α has been chosen, the residence time in state s is exponentially distributed with rate $r^\alpha(s)$. Hence, the probability to leave state s via action α in the time interval $[l, u]$ is given by $\int_l^u r^\alpha(s) \cdot e^{-r^\alpha(s) \cdot t}\, dt$ and the average sojourn time in s is given by $\frac{1}{r^\alpha(s)}$. We say that there is an α-transition from s to s' whenever $\mathbf{P}^\alpha(s, s') \cdot r_\alpha(s) > 0$ where $\mathbf{P}^\alpha(s, s')$ is shorthand of $\mathbf{P}(s, \alpha, s')$. If multiple outgoing α-transitions exist, they compete: the probability that transition $s \xrightarrow{\alpha} s'$ is taken is $\mathbf{P}^\alpha(s, s')$. Putting the pieces together, this means that the CTMDP transits from state s to s' on selecting α in s in the time interval $[l, u]$ with a likelihood that is given by:

$$\mathbf{P}^\alpha(s, s') \cdot \int_l^u r^\alpha(s) \cdot e^{-r^\alpha(s) \cdot t}\, dt.$$

Note that the probabilistic behavior of a CTMDP conforms to that of a CTMC; indeed, if $Act(s)$ is a singleton set in each state $s \in S$, the CTMDP is in fact a CTMC. In this case, the selection of actions is uniquely determined, and the function $\mathbf{P}$ can be projected to an $(S \times S)$-matrix, the transition probability matrix. If we abstract from the exponential state residence times, we obtain a classical MDP. For CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$, its *embedded* MDP is given by $emb(\mathcal{C}) = (S, s_0, Act, \mathbf{P}, L)$.

Example 1. Fig. 2 shows an example CTMDP with AP $= \{a, b\}$ and initial state s_0. The state-labelings are indicated at the states, whereas the transition probabilities are attached to the edges. Rates are omitted from the figure and are defined as: $r^{\alpha}(s_0) = 10$, $r^{\beta}(s_0) = 5$, and $r^{\beta}(s_3) = r^{\beta}(s_1) = r^{\gamma}(s_2) = 1$. In s_0, there is a nondeterministic choice between the actions α and β.

Definition 2 (CTMDP paths). *A sequence* $\pi = s_0 \xrightarrow{\alpha_0, t_0} s_1 \xrightarrow{\alpha_1, t_1} \cdots$ *is an* infinite *path in a* CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$*, where for each* $i \geq 0$*,* $s_i \in S$ *is a state,* $\alpha_i \in Act$ *is an action, and* $t_i \in \mathbb{R}_{>0}$ *is the sojourn time in state* s_i*. A finite path is a fragment of an infinite path ending in a state.*

The length of an infinite path π, denoted $|\pi|$, is ∞; the length of finite path π with $n{+}1$ states is n. For a finite path $\pi = s_0 \xrightarrow{\alpha_0, t_0} s_1 \xrightarrow{\alpha_1, t_1} \cdots \xrightarrow{\alpha_{n-1}, t_{n-1}} s_n$, let $\pi{\downarrow} = s_n$ be the last state of π. Let $Paths(\mathcal{C})$ (respectively $Paths_s(\mathcal{C})$) denote the set of infinite paths (respectively starting in state s) in $\mathcal{C}$; let $Paths^n(\mathcal{C})$ (respectively $Paths^n_s(\mathcal{C})$) denote the set of finite paths of length n (respectively starting in state s). To simplify notation, we omit the reference to $\mathcal{C}$ whenever possible. An example path in the CTMDP of Fig. 2 is $\pi = s_0 \xrightarrow{\alpha, 2.5} s_2 \xrightarrow{\gamma, 1.4} s_0 \xrightarrow{\alpha, \sqrt{2}} s_1 \xrightarrow{\beta, 2.8} s_1 \cdots$.

In order to construct a measurable space over $Paths(\mathcal{C})$, we define the following sets: $\Omega = Act \times \mathbb{R}_{\geq 0} \times S$ and the σ-field $\mathcal{J} = \sigma(2^{Act} \times \mathcal{J}_R \times 2^S)$, where $\mathcal{J}_R$ is the Borel σ-field over $\mathbb{R}_{\geq 0}$. The σ-field over $Paths^n$ is defined as $\mathcal{J}_{Paths^n} = \sigma(\{S_0 \times M_0 \times S_1 \times \cdots \times M_{n-1} \mid S_i \subseteq S, M_i \in \mathcal{J}\})$. A set $B \in \mathcal{J}_{Paths^n}$ is a base of a *cylinder set* C if $C = Cyl(B) = \{\pi \in Paths \mid \pi[0 \ldots n] \in B\}$, where $\pi[0 \ldots n]$ is the prefix of length n of the path π. The σ-field $\mathcal{J}_{Paths}$ of measurable subsets of $Paths(\mathcal{C})$ is defined as $\mathcal{J}_{Paths} = \sigma(\cup_{n=0}^{\infty} \{Cyl(B) \mid B \in \mathcal{J}_{Paths^n}\})$. Hence we obtain a measurable space $(Paths(\mathcal{C}), \mathcal{J}_{Paths})$.

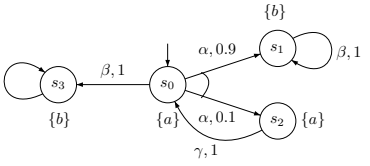


Fig. 2. An example CTMDP

Schedulers. Nondeterminism in a CTMDP is resolved by a *scheduler*. In the literature, schedulers are sometimes also referred to as adversaries, policies, or strategies. For deciding which of the next actions to take, a scheduler may "have access" to the current state only or to the path from the initial to the current state (either with all or with partial information). Schedulers may select the next action either *deterministically*, i.e., depending on the available information, the next action is chosen in a deterministic way, or *randomly*, i.e., depending on the available information, the next action is chosen probabilistically. In our setting, deterministic schedulers suffice to achieve extremum probabilities and can base their decision on a complete information of the current path so far. Moreover, it is not evident how to define the probability measure for randomized schedulers, as exit rates depend on the actions. Hence we only consider deterministic rather than randomized schedulers in this paper. Furthermore, like in [35], we consider measurable functions as schedulers. Formally,

Definition 3 (Schedulers). *A scheduler for* CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$ *is a* measurable *function* $D : Paths(\mathcal{C}) \to Act$ *such that for* $n \in \mathbb{N}$,

$$D(s_0 \xrightarrow{\alpha_0,t_0} s_1 \xrightarrow{\alpha_1,t_1} \cdots \xrightarrow{\alpha_{n-1},t_{n-1}} s_n) \in Act(s_n). \tag{1}$$

We denote the set of all schedulers of $\mathcal{C}$ as $\mathcal{D}_{\mathcal{C}}$.

Remark 1. According to the above definition, we consider schedulers that make a decision as soon as a state is entered. In particular, the sojourn time in the current state s_n is not considered for selecting the next action. Such schedulers are called *early* schedulers in [30]. In contrast, a *late* scheduler will choose an action upon leaving a state, i.e., besides the history $s_0 \xrightarrow{\alpha_0,t_0} \cdots \xrightarrow{\alpha_{n-1},t_{n-1}} s_n$, it will consider also the elapsed time so far in state s_n. Late schedulers suffice for determining extremum reachability probabilities for a certain class of CTMDPs, the so-called locally uniform ones, i.e., CTMDPs in which the exit rate for any enabled action in a state is the same [30].

Probability measure. For a path $\pi \in Paths(\mathcal{C})$ and $m \in \Omega = Act \times \mathbb{R}_{\geq 0} \times S$, we define the *concatenation* of π and m as the path $\pi' = \pi \circ m$. Below we define a probability measure over the measurable space $(Paths(\mathcal{C}), \mathcal{J}_{Paths})$ under the scheduler D.

Definition 4 (Probability measure). *Let* $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$ *be a* CTMDP, $n \in \mathbb{N}$ *and* D *a scheduler in* $\mathcal{D}_{\mathcal{C}}$. *The probability* $\Pr^n_{s,D} : \mathcal{J}_{Paths^n} \to [0, 1]$ *of sets of paths of length* $n > 0$ *starting in* s *is defined inductively by:*

$$\Pr^{n+1}_{s,D}(B) = \int_{Paths^n} \Pr^n_{s,D}(d\pi) \int_{\Omega} \mathbf{1}_B(\pi \circ m) \int_{\mathbb{R}_{\geq 0}} r^{\alpha}(\pi\downarrow) \cdot e^{-r^{\alpha}(\pi\downarrow)\cdot\tau} \cdot \sum_{s' \in S} \mathbf{1}_m(\alpha, \tau, s') \cdot \mathbf{P}^{\alpha}(\pi\downarrow, s')\, dm\ d\tau,$$

where

- $\alpha = D(\pi)$, *the action selected by scheduler* D *on the path* π *of length* n,
- $B \in Paths^{n+1}$ *and for* $n = 0$ *we define* $\Pr^0_s(B) = 1$ *if* $s \in B$, *and* 0 *otherwise,*
- $\mathbf{1}_B(\pi \circ m) = 1$ *when* $\pi \circ m \in B$, *and* 0 *otherwise,*
- $\mathbf{1}_m(\alpha, \tau, s') = 1$ *when* $m = (\alpha, \tau, s')$, *and* 0 *otherwise.*

Intuitively, $\Pr^{n+1}_{s,D}(B)$ is the probability of the set of paths $\pi' = \pi \circ m$ of length $n+1$ defined as a product between the probability of the set of paths π of length n and the one-step transition probability to go from state $\pi\downarrow$ to state $\pi'\downarrow$ by the action α as selected by the scheduler D. For a measurable base $B \in \mathcal{J}_{Paths^n_s}$ and cylinder set $C = Cyl(B)$, let $\Pr_{s,D}(C) = \Pr^n_{s,D}(B)$ as the probability of subsets of paths from $Paths_s$. Sometimes we write $\Pr_D(C)$ to when the starting state s is clear from the context.

2.2 Single-Clock DTA

Let x be a *clock*, which is a variable in $\mathbb{R}_{\geq 0}$[3]. A *clock valuation* is a function η assigning to x the value $\eta(x) \in \mathbb{R}_{\geq 0}$. A *clock constraint* on x is a conjunction of expressions of the form $x \bowtie c$, where $\bowtie \in \{<, \leq, >, \geq\}$ is a binary comparison operator and $c \in \mathbb{N}$. Let $\mathcal{B}_x$ denote the set of clock constraints over x and let g range over $\mathcal{B}_x$.

Definition 5 (DTA). *A single-clock* deterministic timed automaton (DTA) *is a tuple* $\mathcal{A} = (\Sigma, Q, q_0, Q_F, \rightarrow)$ *where*
- Σ *is a finite* alphabet*;*
- Q *is a nonempty finite set of* locations*;*
- $q_0 \in Q$ *is the* initial location*;*
- $Q_F \subseteq Q$ *is a set of* accepting locations*; and*
- $\rightarrow \in (Q \setminus Q_F) \times \Sigma \times \mathcal{B}_x \times \{\varnothing, \{x\}\} \times Q$ *is an* edge relation *satisfying:* $q \xrightarrow{a,g,X} q'$ *and* $q \xrightarrow{a,g',X'} q''$ *with* $g \neq g'$ *implies* $g \wedge g' \equiv \textsc{False}$.

We refer to $q \xrightarrow{a,g,X} q'$ as an *edge*, where $a \in \Sigma$ is an input symbol, the *guard* g is a clock constraint on x, $X = \{\varnothing, \{x\}\}$ is the set of clocks that are to be reset and q' is the successor location. Intuitively, the edge $q \xrightarrow{a,g,X} q'$ asserts that the DTA $\mathcal{A}$ can move from location q to q' when the input symbol is a and the guard g on clock x holds, while the clocks in X should be reset when entering q'. DTA are *deterministic* as they have a single initial location, and outgoing edges of a location labeled with the same input symbol are required to have disjoint guards. In this way, the next location is uniquely determined for a given location and a given clock valuation, together with an action. In case no guard is satisfied in a location for a given clock valuation, time can progress. If the advance of time will never reach a situation in which a guard holds, the DTA will stay in that location ad infinitum. Note that DTA do not have location invariants, as in safety timed automata. However, all the results presented in this paper can be adapted to DTA with invariants without any difficulties.

Runs of a DTA are timed paths. In order to define these formally, we need the following notions on clock valuations. A clock valuation η *satisfies* clock constraint $x \bowtie c$, denoted $\eta \models x \bowtie c$, if and only if $\eta(x) \bowtie c$; it satisfies a conjunction of such expressions if and only if η satisfies all of them. Let $\mathbf{0}$ denote the valuation that assigns 0 to x. The reset of x, denoted $\eta[x := 0]$, is the valuation $\mathbf{0}$. For $\delta \in \mathbb{R}_{\geq 0}$ and η, $\eta{+}\delta$ is the clock-valuation η'' such that $\eta''(x) := \eta(x){+}\delta$.

Definition 6 (Finite DTA path). *A* finite timed path *in* DTA $\mathcal{A}$ *is of the form* $\theta = q_0 \xrightarrow{a_0,t_0} q_1 \xrightarrow{a_1,t_1} \cdots \xrightarrow{a_n,t_n} q_{n+1}$*, such that for all* $0 \leqslant i \leq n$*, it holds* $t_i > 0$*,* $x_0 = 0$*,* $x_j{+}t_j \models g_j$ *and* $x_{j+1} = (x_j{+}t_j)[X_j := 0]$*, where* x_j *is the clock evaluation*[4] *on* entering q_j*,* g_j *is the guard on the uniquely enabled edge in*

[3] Throughout this paper, we use x for the clock variable of the 1-clock DTA under consideration.

[4] As there is only a single clock we sometimes write x for the value of clock x as shorthand for $\eta(x)$.

the DTA *leading from* q_j *to* q_{j+1} *when* $x_j + t_j \models g_j$*, and* X_j *is the set of clocks on that edge that needs to be reset. Path* θ *is* accepted *whenever* $q_{n+1} \in Q_F$*.*

The concepts defined on CTMDP paths, such as $|\theta|$, will be applied to timed DTA paths without modification.

Regions. We consider a variant of the standard region construction for timed automata [2] to DTA. As we consider single-clock DTA, the region construction is rather simple. We basically follow the definition and terminology of [27]. Let $\{c_0, \ldots, c_m\}$ be the set of constants appearing in the guards of DTA $\mathcal{A}$ with $c_0 = 0$. W.l.o.g. we assume $0 = c_0 < c_1 < \cdots < c_m$. Regions can thus be represented by the intervals: $[c_0, c_0], (c_0, c_1), \ldots, [c_m, c_m]$ and (c_m, ∞). (In fact, these regions are also sometimes called zones.) In the continuous probabilistic setting of this paper, the probability of the CTMC taking a transition in a point interval is zero. We therefore combine a region of the form $[c_i, c_i]$ with a region of the form (c_i, c_{i+1}) yielding $[c_i, c_{i+1})$. In the rest of the paper, we slight abuse nomenclature and refer to $[c_i, c_{i+1})$ as a region. As a result, we obtain the regions: $\Theta_0 = [c_0, c_1), \ldots, \Theta_m = [c_m, \infty)$. Let $\Delta c_i = c_{i+1} - c_i$ for $0 \leqslant i < m$ and let $\mathcal{R}_\mathcal{A}$ be the set of regions of DTA $\mathcal{A}$, i.e., $\mathcal{R}_\mathcal{A} = \{\Theta_i \mid 0 \leq i \leq m\}$. The region Θ satisfies a guard g, denoted $\Theta \models g$, iff for all $\eta \in \Theta$ we have $\eta \models g$.

Definition 7 (Region graph). *The* region graph *of* DTA $\mathcal{A} = (\Sigma, Q, q_0, Q_F, \rightarrow)$*, denoted* $\mathcal{G}(\mathcal{A})$*, is the tuple* $(\Sigma, W, w_0, W_F, \dashrightarrow)$ *with* $W = Q \times \mathcal{R}_\mathcal{A}$ *the set of states;* $w_0 = (q_0, \mathbf{0})$ *the initial state;* $W_F = Q_F \times \mathcal{R}_\mathcal{A}$ *the set of final states; and* $\dashrightarrow \subset W \times ((\Sigma \times \{\varnothing, \{x\}\}) \uplus \{\delta\}) \times W$ *the smallest relation such that:*

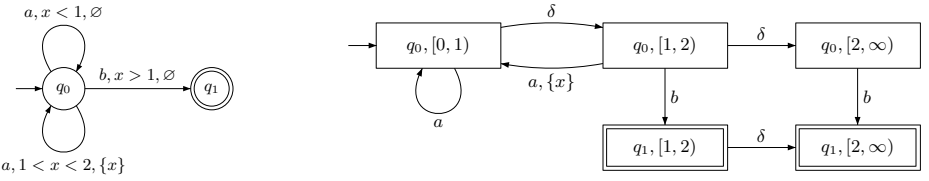
- $(q, \Theta_i) \overset{\delta}{\dashrightarrow} (q, \Theta_{i+1})$ *for* $0 \leq i < m$*;*
- $(q, \Theta_i) \overset{a,\{x\}}{\dashrightarrow} (q', \Theta_0)$ *if* $\exists g \in \mathcal{B}_x$ *such that* $q \xrightarrow{a,g,\{x\}} q'$ *with* $\Theta_i \models g$*; and*
- $(q, \Theta_i) \overset{a,\varnothing}{\dashrightarrow} (q', \Theta_i)$ *if* $\exists g \in \mathcal{B}_x$ *such that* $q \xrightarrow{a,g,\varnothing} q'$ *with* $\Theta_i \models g$*.*

States in $\mathcal{G}(\mathcal{A})$ are thus pairs of locations (of the DTA $\mathcal{A}$) and a region on clock x. The initial state is the initial location in which clock x equals zero. The transition relation of $\mathcal{G}(\mathcal{A})$ is defined using two cases: (1) a delay transition in which the location stays the same, and the region Θ_i is exchanged by its direct successor Θ_{i+1}, (2) a transition that corresponds to taking an enabled edge in the DTA $\mathcal{A}$. The latter corresponds to the last two items in the above definition distinguishing the case in which x is reset (second item) or not (third item).

Example 2. Fig. 3(a) depicts an example DTA, where q_0 is the initial state and q_1 is the only accepting state. In q_0, the guards of the two a-actions are disjoint, so this TA is indeed deterministic. The part of the region graph of the DTA that is reachable from $(q_0, \mathbf{0})$ is depicted in Fig. 3(b).

2.3 Problem Statement

We now are settled to formalize the problem of interest in this paper. Recall that our focus is on using DTA as specification objectives and CTMDPs as system



(a) An example DTA (b) Region graph

Fig. 3. Example DTA and its region graph

models, and our aim is to determine the probability of the set of timed paths of the CTMDP $\mathcal{C}$ that are accepted by $\mathcal{A}$. Let us first define what it means for a CTMDP path to be accepted by DTA $\mathcal{A}$.

Definition 8 (Acceptance). *Given a* CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$ *and a single-clock* DTA $\mathcal{A} = (\Sigma, Q, q_0, Q_F, \rightarrow)$*, we say that an infinite timed path* $\pi = s_0 \xrightarrow{\alpha_0, t_0} s_1 \xrightarrow{\alpha_1, t_1} \cdots$ *in* $\mathcal{C}$ *is* accepted *by* $\mathcal{A}$ *if there exists some* $n \in \mathbb{N}$ *such that the finite fragment of* π *up to* n*, i.e.,* $s_0 \xrightarrow{\alpha_0, t_0} s_1 \cdots s_{n-1} \xrightarrow{\alpha_0, t_{n-1}} s_n$*, gives rise to an "*augmented*" timed path* $\theta = q_0 \xrightarrow{L(s_0), t_0} q_1 \cdots q_{n-1} \xrightarrow{L(s_{n-1}), t_{n-1}} q_n$ *of* $\mathcal{A}$ *with* $q_n \in Q_F$*. Let* $Paths_{s_0}(\mathcal{C} \models \mathcal{A})$ *denote the set of paths in* CTMDP $\mathcal{C}$ *that start in* s_0 *and are accepted by* $\mathcal{A}$*.*

Note that the labels of the states that are visited along the CTMDP path π are used as input symbols for the associated timed path in the DTA. Thus, the alphabet of the DTA will be the powerset of AP, the set of atomic propositions. The aim of this paper is to determine the maximum probability of $Paths_{s_0}(\mathcal{C} \models \mathcal{A})$ over all possible schedulers, i.e.,

$$\sup_{D \in \mathcal{D}_{\mathcal{C}}} \Pr_{s_0, D}(Paths_{s_0}(\mathcal{C} \models \mathcal{A})).$$

In the remainder of this paper, we will show that these maximum probabilities can be characterized as a solution of an LP problem, whose coefficients are given as timed reachability probabilities in a set of CTMDPs. Let us first briefly recall such reachability probabilities.

3 Timed Reachability in CTMDP

Given a CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$, a set of goal states $G \subseteq S$, and a time bound $T \in \mathbb{R}_{\geq 0}$, let $Paths_{s_0}(\Diamond^{\leq T} G)$ denote the set of timed paths reaching G from the initial state s_0 within T time units. Formally,

$$Paths_{s_0}(\Diamond^{\leq T} G) \;=\; \{\pi \in Paths(s_0) \mid \exists t \leq T.\, \pi@t \in G\}$$

where $\pi@t$ denotes the state occupied by π at time t, i.e., $\pi@t = \pi[i]$ where i is the smallest index i such that $\sum_{j=0}^{i} t_j > t$. The timed reachability problem amounts to computing

$$\sup_{D \in \mathcal{D}_{\mathcal{C}}} \Pr_{s_0,D}(Paths_{s_0}(\Diamond^{\leq T} G)).$$

This problem has been solved, to a large extent, forty years ago by Miller [29], and has recently been revisited in the setting of formal verification by, amongst others, [5,31]. We briefly recapitulate the main results. Let $\Psi(s,x)$ be the maximum probability to reach G, within T time units, starting from state s given that x time units have passed so far. It follows that $\Psi(s,x)$ can be characterized by the following set of Bellman equations:

$$\Psi(s,x) = \max_{\alpha \in Act(s)} \left\{ \int_0^{T-x} \sum_{s' \in S} r^{\alpha}(s) \cdot e^{-r^{\alpha}(s) \cdot \tau} \cdot \mathbf{P}^{\alpha}(s,s') \cdot \Psi(s', x+\tau) \; d\tau \right\},$$

if $s \notin G$ and $x \leq T$; and 1 if $s \in G$ and $x \leq T$; and 0, otherwise. The term on the right-hand side takes the action that maximizes the probability to reach G in the remaining $T-x$ time units from s by first moving to s' after a delay of τ time units in s and then proceeding from s' to reach G with elapsed time $x+\tau$.

There are different ways to solve this Bellman equation. One straightforward way is by applying discretization [28,31,17]. An alternative approach is to reduce it to a system of ordinary differential equations (ODEs) with decisions. To that end, let $P_{i,j}(t)$ be the maximum probability to reach state s_j at time t starting from state s_i at time 0. For any two states s_i and s_j we obtain the ODE [8]:

$$\frac{dP_{i,j}(t)}{dt} = \max_{\alpha \in Act(s_i)} \left\{ r^{\alpha}(s_i) \cdot \sum_{s_k \in S} \mathbf{P}^{\alpha}(s_i, s_k) \cdot (P_{k,j}(t) - P_{i,j}(t)) \right\}.$$

which using $\mathbf{R}^{\alpha}(s,s') = r^{\alpha}(s) \cdot \mathbf{P}^{\alpha}(s,s')$ can be simplified to:

$$\frac{dP_{i,j}(t)}{dt} = \max_{\alpha \in Act(s_i)} \left\{ \sum_{s_k \in S} \mathbf{R}^{\alpha}(s_i, s_k) \cdot (P_{k,j}(t) - P_{i,j}(t)) \right\}.$$

For $t \leqslant T$, we obtain the following system of ODEs in matrix form:

$$\frac{d\mathbf{\Pi}(t)}{dt} = \max_{\alpha \in Act} \{\mathbf{\Pi}(t) \cdot \mathbf{Q}^{\alpha}\},$$

where $\mathbf{\Pi}(t)$ is the transition probability matrix at time t, i.e., the element (i,j) of $\mathbf{\Pi}(t)$ equals $P_{i,j}(t)$, $\mathbf{\Pi}(0) = \mathbf{I}$, the identity matrix, $\mathbf{Q}^{\alpha} = \mathbf{R}^{\alpha} - \mathbf{r}^{\alpha}$ is the infinitesimal generator matrix for action α where $\mathbf{R}^{\alpha}$ is the transition rate matrix, i.e., the element (i,j) is $r^{\alpha}(s_i) \cdot \mathbf{P}^{\alpha}(s_i, s_j)$, and $\mathbf{r}^{\alpha}$ is the exit rate matrix in which all diagonal elements are the exit rates, i.e., $\mathbf{r}^{\alpha}(i,i) = r^{\alpha}(s_i)$ and its off-diagonal elements are all zero. Recently, [13] showed that the above system of ODEs can be solved by adopting a technique known as adaptive uniformization.

4 Product Construction

Recall that our aim is to compute the maximum probability of the set of paths of CTMDP $\mathcal{C}$ accepted by the DTA $\mathcal{A}$, that is,

$$\sup_{D \in \mathcal{D}_{\mathcal{C}}} \Pr_{s_0,D}(Paths_{s_0}(\mathcal{C} \models \mathcal{A})).$$

In this section, we show that this can be accomplished by computing maximum reachability probabilities in $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$, i.e., the product between $\mathcal{C}$ and the region graph of $\mathcal{A}$.

Definition 9 (Product). *The product of* CTMDP $\mathcal{C} = (S, s_0, Act, \mathbf{P}, r, L)$ *and* DTA *region graph* $\mathcal{G}(\mathcal{A}) = (\Sigma, W, w_0, W_F, \dashrightarrow)$*, denoted* $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$*, is the tuple* $(Act, V, v_0, V_F, \Lambda, \hookrightarrow)$ *with* $V = S \times W$*,* $v_0 = (s_0, w_0)$*,* $V_F = S \times W_F$*, and*

- $\hookrightarrow \subseteq V \times ((Act \times [0,1] \times \{\varnothing, \{x\}\}) \uplus \{\delta\}) \times V$ *is the smallest relation s.t.:*
  - $(s,w) \overset{\delta}{\hookrightarrow} (s,w')$ iff $w \overset{\delta}{\dashrightarrow} w'$*; and*
  - $(s,w) \overset{\alpha,p,X}{\hookrightarrow} (s',w')$ iff $p = \mathbf{P}^{\alpha}(s,s')$ *with* $p > 0$*, and* $w \overset{L(s),X}{\dashrightarrow} w'$*.*
- $\Lambda : V \times Act \to \mathbb{R}_{\geqslant 0}$ *is the* exit rate function *where:*

$$\Lambda(s,w,\alpha) \;=\; \begin{cases} r^{\alpha}(s) & \text{if } (s,w) \overset{\alpha,p,X}{\hookrightarrow} (s',w') \text{ for some } (s',w') \in V \\ 0 & \text{otherwise.} \end{cases}$$

Example 3. The product of the CTMDP in Fig. 2 and the DTA region graph in Fig. 3(b) is depicted in Fig. 4.

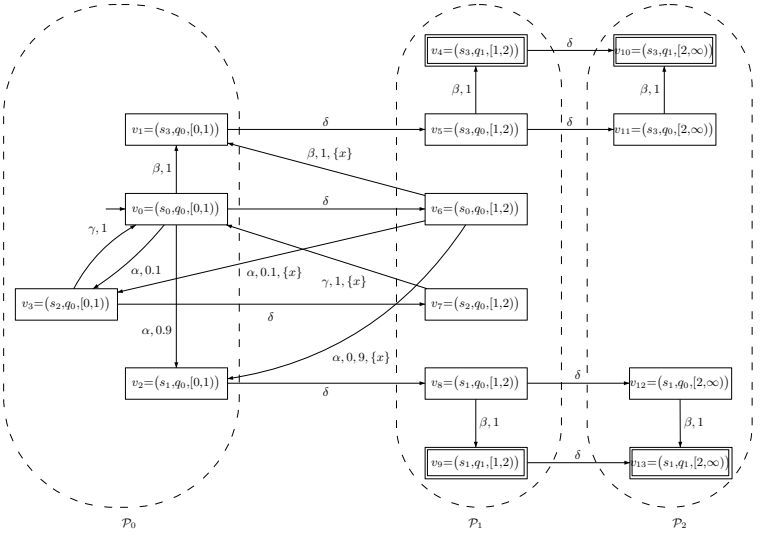


Fig. 4. The product of CTMC and DTA region graph (the reachable part)

Vertex v in the product $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$ is a triple consisting of a CTMDP state s, a DTA state q and a region Θ. Let $v\lfloor_i$ denote the i-th component of the triple v; e.g., if $v = (s,q,\Theta)$, then $v\lfloor_2 = q$. Furthermore, let $Act(v)$ be the set of enabled actions in vertex v, i.e., $Act(v) = Act(v\lfloor_1)$. Edges of the form $v \overset{\delta}{\hookrightarrow} v'$ are called *delay edges*, whereas those of the form $v \overset{\alpha,p,X}{\hookrightarrow} v'$ are called *Markovian edges*.

The product $\mathcal{C} \otimes \mathcal{G}(\mathcal{A})$ is essentially a (simple variant of a) PDDP, i.e., a decision variant of a PDP. The notions of paths and schedulers for a PDDP can be defined in a similar way as for CTMDP in Section 2; we do not dwell upon the details here. For the sake of brevity, let $\mathcal{P} = \mathcal{C} \otimes \mathcal{G}(\mathcal{A})$. In the sequel, let $\mathcal{D}_{\mathcal{P}}$ denote the set of all schedulers of the product $\mathcal{P}$. A scheduler $D \in \mathcal{D}_{\mathcal{P}}$ on the product $\mathcal{P}$ induces a PDP which is equipped with a probability measure $\Pr^{\mathcal{P}}_{v_0,D}$ over its infinite paths in a standard way; for details, we refer to [20]. Let $Paths_{v_0}(\Diamond V_F)$ denote the set of timed paths in $\mathcal{P}$ that reach some vertex in V_F from vertex v_0 starting with clock-valuation $0 \in \Theta_0$.

Lemma 1. *Given* CTMDP $\mathcal{C}$ *and* DTA $\mathcal{A}$*,*

$$\sup_{D \in \mathcal{D}_{\mathcal{C}}} \Pr_{s_0,D}(Paths_{s_0}(\mathcal{C} \models \mathcal{A})) \;=\; \sup_{D \in \mathcal{D}_{\mathcal{P}}} \Pr^{\mathcal{P}}_{v_0,D}(Paths_{v_0}(\Diamond V_F)).$$

Proof (Sketch). We first show that there is a one-to-one correspondence between $Paths_{s_0}(\mathcal{C} \models \mathcal{A})$ and $Paths^{\mathcal{P}}_{s_0}(\Diamond V_F)$.

($\Longrightarrow$) Let $\pi = s_0 \xrightarrow{\alpha_0,t_0} s_1 \cdots s_{n-1} \xrightarrow{\alpha_{n-1},t_{n-1}} s_n$ with $\pi \in Paths_{s_0}(\mathcal{C} \models \mathcal{A})$. We prove that there exists a path $\rho \in Paths^{\mathcal{P}}_{s_0}(\Diamond V_F)$ with $\pi = \rho\downarrow_1$. We have $x_0 = 0$ and for $0 \leq i < n$, $x_i{+}t_i \models g_i$ with $x_{i+1} = (x_i{+}t_1)[X_i := 0]$. Here x_i is the clock valuation in $\mathcal{A}$ on entering state s_i in $\mathcal{C}$. We now construct a timed path θ in $\mathcal{A}$ from π such that $\theta = q_0 \xrightarrow{L(s_0),t_0} q_1 \cdots q_{n-1} \xrightarrow{L(s_{n-1}),t_{n-1}} q_n$, where the clock valuation on entering s_i and q_i coincides. Combining timed paths π and θ yields:

$$\rho = \langle s_0, q_0 \rangle \xrightarrow{t_0} \langle s_1, q_1 \rangle \cdots \langle s_{n-1}, q_{n-1} \rangle \xrightarrow{t_{n-1}} \langle s_n, q_n \rangle,$$

where $\langle s_n, q_n \rangle \in Loc_F$. It follows that $\rho \in Paths^{\mathcal{P}}_{s_0}(\Diamond V_F)$ and $\pi = \rho\downarrow_1$.

($\Longleftarrow$) Let $\rho = \langle s_0, q_0 \rangle \xrightarrow{\alpha_0,t_0} \cdots \xrightarrow{\alpha_{n-1},t_{n-1}} \langle s_n, q_n \rangle \in Paths^{\mathcal{P}}_{s_0}(\Diamond V_F)$. We prove that $\rho\downarrow_1 \in Paths_{s_0}(\mathcal{C} \models \mathcal{A})$. Clearly, we have that $\langle s_n, q_n \rangle \in Loc_F$, $x_0 = 0$, and for $0 \leq i < n$, $x_i{+}t_i \models g_i$ and $x_{i+1} = (x_i{+}t_i)[X_i := 0]$, where x_i is the clock valuation when entering location $\langle s_i, q_i \rangle$. It then directly follows that $q_n \in Q_F$ and $\rho\downarrow_1 \in Paths_{s_0}(\mathcal{C} \models \mathcal{A})$, given the entering clock valuation x_i of state s_i.

Following this path correspondence, it is not difficult to show that for each scheduler D of the CTMDP $\mathcal{C}$, one can construct a scheduler D' of the product $\mathcal{P}$, such that the induced probability measures $\Pr_{s_0,D}$ and $\Pr_{v_0,D'}$ on the corresponding paths coincide. The detailed arguments are quite similar to (and actually simpler than) those of [16, Thm. 4.3]. □

Thanks to this lemma, it suffices to concentrate on determining maximum reachability probabilities in the product $\mathcal{P} = \mathcal{C} \otimes \mathcal{G}(\mathcal{A})$. It is well-known [20] that in this case, memoryless schedulers suffice. This basically stems from the fact that the elapsed time is "encoded" in the state space of the product $\mathcal{P}$; recall that any vertex in $\mathcal{P}$ is of the form (s, q, Θ) where Θ is the current region of the single clock x. Namely, the decision solely depends on (s, q, Θ, x) where (s, q, Θ) is a vertex in $\mathcal{P}$, and x is the actual clock value.

Now we introduce the Bellman equation on the product $\mathcal{P}$ that characterizes $\sup_{D \in \mathcal{D}_{\mathcal{P}}} \Pr^{\mathcal{P}}_{v_0,D}(Paths_{v_0}(\Diamond V_F))$. The following auxiliary definition turns out to

be helpful. For a vertex $v \in V$ with $v\lfloor_3 = \Theta_i$ and clock value x, we define the boundary function $\flat(v,x) = c_{i+1} - x$ if $i < m$; and ∞ if $i = m$. Intuitively, $\flat(v,x)$ is the minimum time (if it exists) to "hit" the boundary of the region of vertex v starting from a clock value x. Let $\Psi(v,x)$ be the maximum probability to reach V_F starting from vertex v given clock value x. Then it follows from [20] that $\Psi(v,x) = 1$ if $v \in V_F$, and otherwise:

$$\Psi(v,x) = \max_{\alpha \in Act(v)} \left\{ \sum_{v \overset{\alpha,p,X}{\hookrightarrow} v'} \int_0^{\flat(v,x)} \underbrace{\Lambda^\alpha(v) \cdot e^{-\Lambda^\alpha(v)\cdot\tau} \cdot p}_{(\star)} \cdot \Psi(v', (x+\tau)[X := 0])\, d\tau \right.$$
$$\left. + \underbrace{\sum_{v \overset{\delta}{\hookrightarrow} v'} e^{-\Lambda^\alpha(v)\cdot\flat(v,x)}}_{(\star\star)} \cdot \Psi\left(v', x + \flat(v,x)\right) \right\}, \tag{2}$$

The term $(\star)$ represents the probability to take the Markovian edge $v \overset{\alpha,p,X}{\hookrightarrow} v'$ while the term $(\star\star)$ denotes the probability to take the delay edge $v \overset{\delta}{\hookrightarrow} v'$. (Note that there is only a single such delay edge, i.e., the second summation ranges over a single delay edge.)

Theorem 1. *For $\mathcal{P} = (Act, V, v_0, V_F, \Lambda, \hookrightarrow)$ we have:*

$$\Psi(v_0, \mathbf{0}) = \sup_{D \in \mathcal{D}_\mathcal{P}} \Pr^{\mathcal{P}}_{v_0,D}(Paths_{v_0}(\Diamond V_F)).$$

Together with Lemma 1, we thus conclude that our problem—determining the maximum probability that CTMDP $\mathcal{C}$ satisfies the DTA specification $\mathcal{A}$—reduces to determining $\Psi(v_0, \mathbf{0})$ for the Bellman equation (2) on the product $\mathcal{P} = \mathcal{C} \otimes \mathcal{G}(\mathcal{A})$.

5 Reduction to a Linear Programming Problem

In this section, we show that the solution $\Psi(v_0, \mathbf{0})$ of the Bellman equation (2) coincides with the solution of an LP problem whose coefficients are maximum timed reachability probabilities in a set of CTMDPs that are obtained by a graph decomposition of the product $\mathcal{P} = \mathcal{C} \otimes \mathcal{G}(\mathcal{A})$. Let us first define the graph decomposition of the product $\mathcal{P}$. The operational intuition can best be explained by means of our running example, see Fig. 4. The idea is to group all vertices with the same region, i.e., we group the vertices in a column-wise manner. In the example this yields three sub-graphs $\mathcal{P}_0$ through $\mathcal{P}_2$. A delay in $\mathcal{P}_i$ (with $i = 0, 1$) yields a vertex in $\mathcal{P}_{i+1}$, taking an edge in the DTA in which clock x is unaffected (i.e., not reset) yields a vertex in $\mathcal{P}_i$ (for all i), whereas in case clock x is reset, a vertex in $\mathcal{P}_0$ is obtained. This is formalized below as follows.

Definition 10 (Graph decomposition). *The* graph decomposition *of $\mathcal{P} = (Act, V, v_0, V_F, \Lambda, \hookrightarrow)$ yields the set of graphs $\{\mathcal{P}_i \mid 0 \leq i \leq m\}$ where $\mathcal{P}_i = (Act, V_i, V_{F_i}, \Lambda_i, \hookrightarrow_i)$ with:*

- $V_i = \{(s, q, \Theta_i) \in V\}$ *and* $V_{F_i} = V_i \cap V_F$,
- $\Lambda_i^\alpha(v) = \Lambda^\alpha(v)$, *for* $v \in V_i$, *and*
- $\hookrightarrow_i = \big(\bigcup_{\alpha \in Act}\{M_i^\alpha \cup B_i^\alpha\}\big) \cup F_i$ *where:*
  - M_i^α *is the set of Markovian edges (without reset) within* $\mathcal{P}_i$ *labeled by* α,
  - B_i^α *(backward) is the set of Markovian edges (with reset) from* $\mathcal{P}_i$ *to* $\mathcal{P}_0$,
  - F_i *(forward) is the set of delay edges from the vertices in* $\mathcal{P}_i$ *to* $\mathcal{P}_{i+1}$.

As the graph $\mathcal{P}_m$ only involves unbounded regions, it has no outgoing delay transitions.

Example 4. The product $\mathcal{P}$ in Fig. 4 is decomposed into the graphs $\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_2$ as indicated by the dashed ovals. For $\mathcal{P}_1$, e.g., we have $M_1^\beta = \{v_5 \hookrightarrow v_4, v_8 \hookrightarrow v_9\}$; $B_1^\alpha = \{v_6 \hookrightarrow v_3, v_6 \hookrightarrow v_2\}$, $B_1^\beta = \{v_6 \hookrightarrow v_1\}$, and $B_1^\gamma = \{v_7 \hookrightarrow v_0\}$. Its delay transitions are $F_1 = \{v_4 \hookrightarrow v_{10}, v_5 \hookrightarrow v_{11}, v_8 \hookrightarrow v_{12}, v_9 \hookrightarrow v_{13}\}$.

For graph $\mathcal{P}_i$ $(0 \le i \le m)$ with $|V_i| = k_i$, define the probability vector

$$\vec{U}_i(x) = [u_i^1(x), \ldots, u_i^{k_i}(x)]^\mathsf{T} \in \mathbb{R}(x)^{k_i \times 1},$$

where $u_i^j(x)$ is the maximum probability to go from vertex $v_i^j \in V_i$ to some vertex in the goal set V_F (in $\mathcal{M}$) at time point x. Our aim is to determine $\vec{U}_0(0)$. In the sequel, we aim to establish a relationship between $\vec{U}_i(0)$ and $\vec{U}_j(0)$ for $i \neq j$. To that end, we distinguish two cases:

Case $0 \le i < m$. We first introduce some definitions.

- $\mathbf{P}_i^{\alpha,\mathsf{M}} \in [0,1]^{k_i \times k_i}$ and $\mathbf{P}_i^{\alpha,\mathsf{B}} \in [0,1]^{k_i \times k_0}$ are probability transition matrices for Markovian and backward transitions respectively, parameterized by action α. For $\alpha \in Act(v)$, let $\mathbf{P}_i^{\alpha,\mathsf{M}}[v, v'] = p$, if $v \overset{\alpha,p,\varnothing}{\hookrightarrow} v'$; and 0 otherwise. Similarly $\mathbf{P}_i^{\alpha,\mathsf{B}}[v, v'] = p$ if $v \overset{\alpha,p,\{x\}}{\hookrightarrow} v'$; and 0 otherwise. Moreover, let $\mathbf{P}_i^\alpha = \big(\mathbf{P}_i^{\alpha,\mathsf{M}} \big| \mathbf{P}_i^{\alpha,\mathsf{B}}\big)$. Note that $\mathbf{P}_i^\alpha$ is a stochastic matrix, as:

$$\sum_{v'} \mathbf{P}_i^{\alpha,\mathsf{M}}[v, v'] + \sum_{v''} \mathbf{P}_i^{\alpha,\mathsf{B}}[v, v''] \;=\; 1.$$

- $\mathbf{D}_i^\alpha(x) \in \mathbb{R}^{k_i \times k_i}$ is the delay probability matrix, i.e., for any $1 \le j \le k_i$, $\mathbf{D}_i^\alpha(x)[j, j] = e^{-r^\alpha(v_i^j)x}$. Its off-diagonal elements are zero;
- $\mathbf{E}_i^\alpha \in \mathbb{R}^{k_i \times k_i}$ is the exit rate matrix, i.e., for any $1 \le j \le k_i$, $\mathbf{E}_i^\alpha[j, j] = r^\alpha(v_i^j)$. Its off-diagonal elements are zero;
- $\mathbf{M}_i^\alpha(x) = \mathbf{E}_i^\alpha \cdot \mathbf{D}_i^\alpha(x) \cdot \mathbf{P}_i^{\alpha,\mathsf{M}} \in \mathbb{R}^{k_i \times k_i}$ is the probability density matrix for Markovian transitions inside $\mathcal{P}_i$. Namely, $\mathbf{M}_i^\alpha(x)[j, j']$ indicates the pdf to take the α-labelled Markovian edge without reset from the j-th vertex to the j'-th vertex in $\mathcal{P}_i$;
- $\mathbf{B}_i^\alpha(x) = \mathbf{E}_i^\alpha \cdot \mathbf{D}_i^\alpha(x) \cdot \mathbf{P}_i^{\alpha,\mathsf{B}} \in \mathbb{R}^{k_i \times k_0}$ is the probability density matrix for the reset edges B_i^α. Namely, $\mathbf{B}_i^\alpha(x)[j, j']$ indicates the pdf to take the Markovian edge with reset from the j-th vertex in $\mathcal{P}_i$ to the j'-th vertex in $\mathcal{P}_0$;

– $\mathbf{F}_i \in \mathbb{R}^{k_i \times k_{i+1}}$ is the incidence matrix for delay edges F_i. Thus, $\mathbf{F}_i[j, j'] = 1$ iff there is a delay edge from the j-th vertex in $\mathcal{P}_i$ to the j'-th vertex in $\mathcal{P}_{i+1}$.

Example 5 (Continuing Example 4). According to the definitions, we have the following matrices for $\mathcal{P}_0$ and $\mathcal{P}_1$. Let r_i^α be a shorthand of the exit rate $r^\alpha(s_i)$:

$$\mathbf{M}_0^\alpha(x) = \underbrace{\begin{pmatrix} r_0^\alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{E}_0} \underbrace{\begin{pmatrix} e^{-r_0^\alpha x} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{D}_0(x)} \underbrace{\begin{pmatrix} 0 & 0 & 0.9 & 0.1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{P}_0^{\alpha,\mathsf{M}}} = \begin{pmatrix} 0 & 0 & 0.9r_0^\alpha e^{-r_0^\alpha x} & 0.1r_0^\alpha e^{-r_0^\alpha x} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Similarly,

$$\mathbf{B}_1^\beta(x) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1{\cdot}e^{-r_0^\beta x} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } \mathbf{F}_0 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

By instantiating (2), we obtain the following for $0 \le i < m$:

$$\vec{U}_i(x) = \max_{\alpha \in Act} \Big\{ \underbrace{\int_0^{\Delta c_i - x} \mathbf{M}_i^\alpha(\tau) \cdot \vec{U}_i(x{+}\tau) d\tau}_{(\star)} + \underbrace{\int_0^{\Delta c_i - x} \mathbf{B}_i^\alpha(\tau)\, d\tau \cdot \vec{U}_0(0)}_{(\star\star)} + \mathbf{D}_i^\alpha(\Delta c_i {-} x) \cdot \mathbf{F}_i \vec{U}_{i+1}(0) \Big\}, \tag{3}$$

Let us explain the above equation. First of all, recall that $\flat(v, x) = \Delta c_i - x$ for each state $v \in V_i$ with $i < m$. Term $(\star)$ (resp. $(\star\star)$) reflects the case where clock x is not reset (resp. is reset and returned to $\mathcal{P}_0$). Note that $\mathbf{M}_i^\alpha(\tau)$ and $\mathbf{B}_i^\alpha(\tau)$ are the matrix forms of the density function $(\star)$ in (2). The matrix $\mathbf{D}_i^\alpha(\Delta c_i - x)$ indicates the probability to delay until the "end" of region i, and $\mathbf{F}_i \cdot \vec{U}_{i+1}(0)$ denotes the probability to continue in $\mathcal{P}_{i+1}$ (at relative time point 0), and $\mathbf{D}_i^\alpha \cdot (\Delta c_i - x) \cdot \mathbf{F}_i$ is the matrix form of the term $(\star\star)$ in (2).

Case $i = m$. In this case, $\vec{U}_m(x)$ is simplified as follows:

$$\vec{U}_m(x) = \max_{\alpha \in Act} \Big\{ \int_0^\infty \widehat{\mathbf{M}}_m^\alpha(\tau) \cdot \vec{U}_m(x{+}\tau)\, d\tau + \tilde{\mathbf{1}}_F + \int_0^\infty \mathbf{B}_m^\alpha(\tau) d\tau \cdot \vec{U}_0(0) \Big\}, \tag{4}$$

where $\widehat{\mathbf{M}}_m^\alpha(\tau)[v, \cdot] = \mathbf{M}_m^\alpha(\tau)[v, \cdot]$ for $v \notin V_F$, 0 otherwise. $\tilde{\mathbf{1}}_F$ is a characteristic vector such that $\tilde{\mathbf{1}}_F[v] = 1$ iff $v \in V_F$.

Our remaining task now is to solve the system of integral equations given by equations (3) and (4). First observe that, due to the fact that $\mathcal{P}_i$ only contains

Markovian edges, the structure (V_i, Λ_i, M_i) forms a CTMDP, referred to as $\mathcal{C}_i$. For each $\mathcal{P}_i$, we define the *augmented* CTMDP $\mathcal{C}_i^\star$ with state space $V_i \cup V_0$ such that all V_0-vertices are made absorbing (i.e., their outgoing edges are replaced by a self-loop) and all edges connecting V_i to V_0 are kept. The augmented CTMDP $\mathcal{C}_1^\star$ for $\mathcal{P}_1$ in Fig. 4 is shown in Fig. 5.

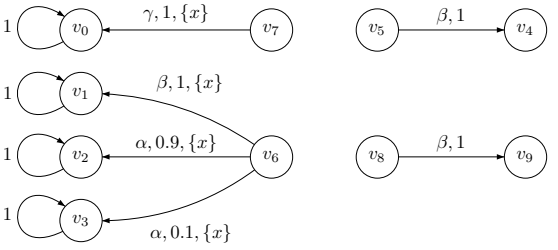


Fig. 5. Augmented CTMDP $\mathcal{C}_1^\star$

By instantiating (2), we have the following equation (in the matrix form) for the transition probability:

$$\mathbf{\Pi}(x) \;=\; \max_{\alpha \in Act} \left\{ \int_0^x \widetilde{\mathbf{M}}^\alpha(\tau) \cdot \mathbf{\Pi}(x-\tau)\, d\tau \right\} + \mathbf{D}^\alpha(x), \tag{5}$$

where $\widetilde{\mathbf{M}}^\alpha(\tau)[v, v'] = r^\alpha(v) \cdot e^{-r^\alpha(v)\cdot\tau} \cdot p$ if there is a Markovian edge $v \overset{\alpha,p,\varnothing}{\hookrightarrow} v'$; 0 otherwise. In fact, the characterization of $\Psi(s, x)$ in Section 4 is an equivalent formulation of Eq.(5). For augmented CTMDP $\mathcal{C}_i^\star$, $\widetilde{\mathbf{M}}^\alpha(\tau)$ we have:

$$\widetilde{\mathbf{M}}^\alpha(\tau) = \left(\begin{array}{c|c} \mathbf{M}_i^\alpha(\tau) & \mathbf{B}_i^\alpha(\tau) \\ \hline \mathbf{0} & \mathbf{I} \end{array} \right),$$

where $\mathbf{0} \in \mathbb{R}^{k_0 \times k_i}$ is the matrix with all 0's and $\mathbf{I} \in \mathbb{R}^{k_0 \times k_0}$ is the identity matrix.

Now given any CTMDP $\mathcal{C}_i$ (resp. augmented CTMDP $\mathcal{C}_i^\star$) corresponding to $\mathcal{P}_i$, we obtain Eq. (5), and write its solution as $\mathbf{\Pi}_i(x)$ (resp. $\mathbf{\Pi}_i^\star(x)$). We then define $\bar{\mathbf{\Pi}}_i^\star \in \mathbb{R}^{k_i \times k_0}$ for an augmented CTMDP $\mathcal{C}_i^\star$ to be part of $\mathbf{\Pi}_i^\star$, where $\bar{\mathbf{\Pi}}_i^\star$ only keeps the probabilities starting from V_i and ending in V_0. As a matter of fact,

$$\mathbf{\Pi}_i^\star(x) = \left(\begin{array}{c|c} \mathbf{\Pi}_i(x) & \bar{\mathbf{\Pi}}_i^\star(x) \\ \hline \mathbf{0} & \mathbf{I} \end{array} \right).$$

The following theorem is the key result of this section. Its proof is technically involved and is given in the Appendix.

Theorem 2. *For sub-graph $\mathcal{P}_i$ of $\mathcal{P}$, it holds:*

$$\vec{U}_i(0) \;=\; \mathbf{\Pi}_i(\Delta c_i) \cdot \mathbf{F}_i \cdot \vec{U}_{i+1}(0) \;+\; \bar{\mathbf{\Pi}}_i^\star(\Delta c_i) \cdot \vec{U}_0(0), \qquad \text{if } 0 \le i < m \tag{6}$$

where $\mathbf{\Pi}_i(\cdot)$ and $\bar{\mathbf{\Pi}}_i^\star(\cdot)$ are defined on CTMDP $\mathcal{C}_i$ *and (augmented) $\mathcal{C}_i^\star$ as above.*

$$\vec{U}_m(0) \;=\; \max_{\alpha \in Act} \left\{ \widehat{\mathbf{P}}_m^\alpha \cdot \vec{U}_m(0) + \vec{\mathbf{1}}_F + \widehat{\mathbf{B}}_m^\alpha \cdot \vec{U}_0(0) \right\}, \qquad \text{if } i = m \tag{7}$$

with $\widehat{\mathbf{P}}_m^\alpha(v, v') = \mathbf{P}_m^\alpha(v, v')$ if $v \notin V_{F_m}$; 0 otherwise, and $\widehat{\mathbf{B}}_m^\alpha = \int_0^\infty \mathbf{B}_m^\alpha(\tau)\, d\tau$.

Recall that we intend to solve the system of integral equations given by the equations (3) and (4) so as to obtain the vectors $\vec{U}_i(0)$ for $0 \leq i \leq m$. Theorem 2 entails that instead of accomplishing this directly, one alternatively can exploit equations 6 and 7, where $\vec{U}_i(0)$ $(0 \leq i \leq m)$ can be regarded as a family of variables and the coefficients $\mathbf{\Pi}_i(\cdot)$ can be obtained by computing the corresponding maximum timed reachability probabilities of CTMDPs $\mathcal{C}_i^\star$. It is not difficult to see that the set of equations in Theorem 2 can be easily reduced to an LP problem, see, e.g., [9].

6 Conclusion

We showed that the verification of CTMDPs against 1-clock DTA objectives can be reduced to solving an LP problem whose coefficients are extremum timed reachability probabilities in CTMDPs. This extends the class of timed reachability properties to an interesting and practically relevant set of properties. The main ingredients of our approach are a region graph and a product construction, computing timed reachability probabilities in a set of CTMDPs, and finally solving an LP problem. The availability of the first practical implementations for timed reachability of CTMDPs paves the way to a realization of our approach in a prototypical tool. Like in [7], our approach facilitates optimizations such as parallelization and bisimulation minimization. Such implementation and experimentation is essential to show the practical feasibility of our approach and is left for further work.

Another interesting research direction is to consider other acceptance criteria for DTA, such as Muller acceptance. We claim that this can basically be done along the lines of [16] for CTMCs; the main technical difficulty is that one needs to resort to either finite memory schedulers or randomized schedulers, see e.g. [14].

References

1. Aceto, L., Bouyer, P., Burgueño, A., Larsen, K.G.: The power of reachability testing for timed automata. Theor. Comput. Sci. 300(1-3), 411–475 (2003)
2. Alur, R., Dill, D.L.: A theory of timed automata. Theor. Comput. Sci. 126(2), 183–235 (1994)
3. Baier, C., Cloth, L., Haverkort, B.R., Kuntz, M., Siegle, M.: Model checking Markov chains with actions and state labels. IEEE Trans. Software Eng. 33(4), 209–224 (2007)
4. Baier, C., Haverkort, B.R., Hermanns, H., Katoen, J.-P.: Model-checking algorithms for continuous-time Markov chains. IEEE Trans. Software Eng. 29(6), 524–541 (2003)
5. Baier, C., Hermanns, H., Katoen, J.-P., Haverkort, B.: Efficient computation of time-bounded reachability probabilities in uniform continuous-time Markov decision processes. Theor. Comput. Sci. 345(1), 2–26 (2005)
6. Baier, C., Kwiatkowska, M.: Model checking for a probabilistic branching time logic with fairness. Distrib. Comput. 11, 125–155 (1998)

7. Barbot, B., Chen, T., Han, T., Katoen, J.-P., Mereacre, A.: Efficient CTMC model checking of linear real-time objectives. In: Abdulla, P.A., Leino, K.R.M. (eds.) TACAS 2011. LNCS, vol. 6605, pp. 128–142. Springer, Heidelberg (2011)
8. Bellman, R.: Dynamic Programming. Princeton University Press, Princeton (1957)
9. Bianco, A., de Alfaro, L.: Model checking of probabilistic and nondeterministic systems. In: Thiagarajan, P.S. (ed.) FSTTCS 1995. LNCS, vol. 1026, pp. 499–513. Springer, Heidelberg (1995)
10. Brázdil, T., Forejt, V., Krcál, J., Kretínský, J., Kucera, A.: Continuous-time stochastic games with time-bounded reachability. In: FSTTCS, pp. 61–72 (2009)
11. Brázdil, T., Krcál, J., Kretínský, J., Kucera, A., Rehák, V.: Stochastic real-time games with qualitative timed automata objectives. In: Gastin, P., Laroussinie, F. (eds.) CONCUR 2010. LNCS, vol. 6269, pp. 207–221. Springer, Heidelberg (2010)
12. Brázdil, T., Krcál, J., Kretínský, J., Kucera, A., Rehák, V.: Measuring performance of continuous-time stochastic processes using timed automata. In: HSCC, pp. 33–42. ACM Press, New York (2011)
13. Buchholz, P., Schulz, I.: Numerical analysis of continuous time Markov decision processes over finite horizons. Computers & OR 38(3), 651–659 (2011)
14. Chatterjee, K., de Alfaro, L., Henzinger, T.A.: Trading memory for randomness. In: Quantitative Evaluation of Systems (QEST), pp. 206–217. IEEE Computer Society, Los Alamitos (2004)
15. Chen, T., Han, T., Katoen, J.-P., Mereacre, A.: Quantitative model checking of continuous-time Markov chains against timed automata specifications. In: LICS, pp. 309–318 (2009)
16. Chen, T., Han, T., Katoen, J.-P., Mereacre, A.: Model checking of continuous-time Markov chains against timed automata specifications. Logical Methods in Computer Science 7(1–2), 1–34 (2011)
17. Chen, T., Han, T., Katoen, J.-P., Mereacre, A.: Reachability probabilities in Markovian timed automata. Technical report, RR-11-02, OUCL (2011)
18. Ciesinski, F., Baier, C.: Liquor: A tool for qualitative and quantitative linear time analysis of reactive systems. In: Quantitative Evaluation of Systems (QEST), pp. 131–132. IEEE Computer Society, Los Alamitos (2006)
19. Courcoubetis, C., Yannakakis, M.: The complexity of probabilistic verification. J. ACM 42(4), 857–907 (1995)
20. Davis, M.H.A.: Markov Models and Optimization. Chapman and Hall, Boca Raton (1993)
21. de Alfaro, L.: How to specify and verify the long-run average behavior of probabilistic systems. In: LICS, pp. 454–465 (1998)
22. Donatelli, S., Haddad, S., Sproston, J.: Model checking timed and stochastic properties with $\mathrm{CSL}^{\mathrm{TA}}$. IEEE Trans. Software Eng. 35(2), 224–240 (2009)
23. Guo, X., Hernández-Lerma, O., Prieto-Rumeau, T.: A survey of recent results on continuous-time Markov decision processes. TOP 14(2), 177–257 (2006)
24. Howard, R.A.: Dynamic Programming and Markov Processes. MIT Press, Cambridge (1960)
25. Katoen, J.-P., Zapreev, I., Hahn, E.M., Hermanns, H., Jansen, D.N.: The ins and outs of the probabilistic model checker MRMC. Perf. Ev. 68(2), 90–104 (2011)
26. Knast, R.: Continuous-time probabilistic automata. Information and Control 15(4), 335–352 (1969)
27. Laroussinie, F., Markey, N., Schnoebelen, P.: Model checking timed automata with one or two clocks. In: Gardner, P., Yoshida, N. (eds.) CONCUR 2004. LNCS, vol. 3170, pp. 387–401. Springer, Heidelberg (2004)

28. Martin-Löfs, A.: Optimal control of a continuous-time Markov chain with periodic transition probabilities. Operations Research 15, 872–881 (1967)
29. Miller, B.L.: Finite state continuous time Markov decision processes with a finite planning horizon. SIAM Journal on Control 6(2), 266–280 (1968)
30. 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)
31. Neuhäußer, M.R., Zhang, L.: Time-bounded reachability probabilities in continuous-time Markov decision processes. In: Quantitative Evaluation of Systems (QEST), pp. 209–218. IEEE Computer Society, Los Alamitos (2010)
32. Puterman, M.L.: Markov Decision Processes. Wiley, Chichester (1994)
33. Rabe, M., Schewe, S.: Finite optimal control for time-bounded reachability in CTMDPs and continuous-time Markov games. CoRR, abs/1004.4005 (2010)
34. Vardi, M.Y.: Automatic verification of probabilistic concurrent finite-state programs. In: FOCS, pp. 327–338. IEEE Computer Society, Los Alamitos (1985)
35. Wolovick, N., Johr, S.: A characterization of meaningful schedulers for continuous-time Markov decision processes. In: Asarin, E., Bouyer, P. (eds.) FORMATS 2006. LNCS, vol. 4202, pp. 352–367. Springer, Heidelberg (2006)

A Proof of Theorem 2

Theorem 2. For subgraph $\mathcal{P}_i$ of $\mathcal{M}$ with k_i states, it holds:

- For $0 \leq i < m$,

$$\vec{U}_i(0) = \mathbf{\Pi}_i(\Delta c_i) \cdot \mathbf{F}_i \vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}_i^{\star}(\Delta c_i) \cdot \vec{U}_0(0), \tag{8}$$

  where $\mathbf{\Pi}_i(\Delta c_i)$ and $\bar{\mathbf{\Pi}}_i^{\star}(\Delta c_i)$ are for CTMDP $\mathcal{C}_i$ and (augmented) $\mathcal{C}_i^{\star}$, respectively.
- For $i = m$,

$$\vec{U}_m(0) = \max_{\alpha \in Act} \left\{ \widehat{\mathbf{P}}_m^{\alpha} \cdot \vec{U}_m(0) + \vec{\mathbf{1}}_F + \widehat{\mathbf{B}}_m^{\alpha} \cdot \vec{U}_0(0) \right\}, \tag{9}$$

  where $\widehat{\mathbf{P}}_m^{\alpha}(v, v') = \mathbf{P}_m^{\alpha}(v, v')$ if $v \notin V_{F_m}$; 0 otherwise, and $\widehat{\mathbf{B}}_m^{\alpha} = \int_0^{\infty} \mathbf{B}_m^{\alpha}(\tau) d\tau$.

Proof. We first deal with the case $i < m$. If in $\mathcal{P}_i$, for some action α there exists some backward edge, namely, for some j, j', $\mathbf{B}_i^{\alpha}(x)[j, j'] \neq 0$, then we shall consider the *augmented* CTMDP $\mathcal{C}_i^{\star}$ with $k_i^{\star} = k_i + k_0$ states. In view of this, the augmented version of the integral equation $\vec{V}_i(x)$ is defined as:

$$\vec{V}_i^{\star}(x) = \max_{\alpha \in Act} \left\{ \int_0^{\Delta c_i - x} \mathbf{M}_i^{\alpha,\star}(\tau) \cdot \vec{V}_i^{\star}(x+\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(\Delta c_i - x) \cdot \mathbf{F}_i^{\star} \cdot \vec{\widehat{V}}_i(0) \right\},$$

where

- $\vec{V}_i^{\star}(x) = \begin{pmatrix} \vec{V}_i(x) \\ \vec{V}_i'(x) \end{pmatrix} \in \mathbb{R}^{k_i^{\star} \times 1}$, where $\vec{V}_i'(x) \in \mathbb{R}^{k_0 \times 1}$ is the vector representing reachability probabilities for the augmented states in $\mathcal{P}_i$;

- $\mathbf{M}_i^{\alpha,\star}(\tau) = \left(\begin{array}{c|c} \mathbf{M}_i^{\alpha}(\tau) & \mathbf{B}_i^{\alpha}(\tau) \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right) \in \mathbb{R}^{k_i^\star \times k_i^\star}$. The exit rate of augmented states is 0 for all actions.
- $\mathbf{D}_i^{\alpha,\star}(\tau) = \left(\begin{array}{c|c} \mathbf{D}_i^{\alpha}(\tau) & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \in \mathbb{R}^{k_i^\star \times k_i^\star}$.
- $\mathbf{F}_i^\star = \left(\mathbf{F}_i' \middle| \mathbf{B}_i'\right) \in \mathbb{R}^{k_i^\star \times (k_{i+1}+k_0)}$ such that
  - $\mathbf{F}_i' = \left(\frac{\mathbf{F}_i}{\mathbf{0}}\right) \in \mathbb{R}^{k_i^\star \times k_{i+1}}$ is the incidence matrix for delay edges and
  - $\mathbf{B}_i' = \left(\frac{\mathbf{0}}{\mathbf{I}}\right) \in \mathbb{R}^{k_i^\star \times k_0}$.
- $\vec{\widehat{V}}_i(0) = \left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \in \mathbb{R}^{(k_{i+1}+k_0)\times 1}$.

In the sequel, we prove two claims:

Claim 1. For each $0 \le j \le k_i$, $\vec{U}_i[j] = \vec{V}_i^\star[j]$.

Proof of Claim 1. According to the definition, we have that

$$\vec{V}_i^\star(x) = \max_{\alpha \in Act} \left\{ \int_0^{\Delta c_i - x} \left(\begin{array}{c|c} \mathbf{M}_i^{\alpha}(\tau) & \mathbf{B}_i^{\alpha}(\tau) \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right) \cdot \vec{V}_i^\star(x+\tau) d\tau + \left(\begin{array}{c|c} \mathbf{D}_i^{\alpha}(\Delta c_i - x) & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \cdot \left(\begin{array}{c|c} \mathbf{F}_i & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \cdot \left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \right\}.$$

It follows immediately that $\vec{V}_i'(x) = \vec{U}_0(0)$. For $\vec{V}_i(x)$, we have that

$$\begin{aligned} & \vec{V}_i(x) \\ = & \max_{\alpha \in Act} \Big\{ \int_0^{\Delta c_i - x} \mathbf{M}_i^{\alpha}(\tau) \vec{V}_i(x+\tau) d\tau + \int_0^{\Delta c_i - x} \mathbf{B}_i^{\alpha}(\tau) \vec{V}_i'(x+\tau) d\tau \\ & \qquad + \mathbf{D}_i^{\alpha}(\Delta c_i - x) \cdot \mathbf{F}_i \cdot \vec{U}_{i+1}(0) \Big\} \\ = & \max_{\alpha \in Act} \Big\{ \int_0^{\Delta c_i - x} \mathbf{M}_i^{\alpha}(\tau) \vec{V}_i(x+\tau) d\tau + \int_0^{\Delta c_i - x} \mathbf{B}_i^{\alpha}(\tau) d\tau \cdot \vec{U}_0(0) \\ & \qquad + \mathbf{D}_i^{\alpha}(\Delta c_i - x) \cdot \mathbf{F}_i \cdot \vec{U}_{i+1}(0) \Big\} \\ = & \vec{U}_i(x) \ . \end{aligned}$$

♣

Claim 2.

$$\vec{V}_i^\star(x) = \mathbf{\Pi}_i^\star(\Delta c_i - x) \cdot \mathbf{F}_i^\star \vec{\widehat{V}}_i(0) \ ,$$

where

$$\mathbf{\Pi}_i^\star(x) = \max_{\alpha \in Act} \left\{ \int_0^x \mathbf{M}_i^{\alpha,\star}(\tau) \mathbf{\Pi}_i^\star(x-\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(x) \right\} \ .$$

Proof of Claim 2. Standard arguments yield that the optimal probability corresponds to the least fixpoint of a functional and can be computed iteratively. Let $c_{i,x} = \Delta c_i - x$. We define

$$\vec{V}_i^{\star,(0)}(x) = \vec{0}$$
$$\vec{V}_i^{\star,(j+1)}(x) = \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\alpha}(\tau) \vec{V}_i^{\star,(j)}(x+\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \right\} \ .$$

and

$$\mathbf{\Pi}_i^{\star,(0)}(c_{i,x}) = \mathbf{0}$$
$$\mathbf{\Pi}_i^{\star,(j+1)}(c_{i,x}) = \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\star}(\tau) \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}-\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(c_{i,x}) \right\} \ .$$

By induction on j, we prove the following relation:

$$\vec{V}_i^{\star,(j)}(x) = \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}) \cdot \mathbf{F}_i \vec{\widehat{V}}_i(0) \ .$$

- Base case. $\vec{V}_i^{\star,(0)}(x) = \vec{0}$ and $\mathbf{\Pi}_i^{\star,(0)}(c_{i,x}) = \mathbf{0}$.
- Induction hypothesis.

$$\vec{V}_i^{\star,(j)}(x) = \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{U}}_i(0) \ .$$

- Induction step. We have that

$$\vec{V}_i^{\star,(j+1)}(x) = \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\star,\alpha}(\tau) \vec{V}_i^{\star,(j)}(x+\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{U}}_i(0) \right\}.$$

It follows that

$$\begin{aligned}
&\vec{V}_i^{\star,(j+1)}(x) \\
&= \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\star,\alpha}(\tau) \vec{V}_i^{\star,(j)}(x+\tau) d\tau + \mathbf{D}_i^{\alpha,\star}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \right\} \\
&\overset{\text{I.H.}}{=} \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\star,\alpha}(\tau) \cdot \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}-\tau) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) d\tau + \mathbf{D}_i^{\alpha,\star}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \right\} \\
&= \max_{\alpha \in Act} \left\{ \left(\int_0^{c_{i,x}} \mathbf{M}_i^{\star,\alpha}(\tau) \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}-\tau) d\tau + \mathbf{D}_i^{\star}(c_{i,x}) \right) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \right\} \\
&= \max_{\alpha \in Act} \left\{ \int_0^{c_{i,x}} \mathbf{M}_i^{\star,\alpha}(\tau) \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}-\tau) d\tau + \mathbf{D}_i^{\alpha} \star (c_{i,x}) \right\} \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \\
&= \mathbf{\Pi}_i^{\alpha,(j+1)}(c_{i,x}) \cdot \mathbf{F}_i^{\star} \vec{\widehat{V}}_i(0) \ .
\end{aligned}$$

Clearly,

$$\mathbf{\Pi}_i^{\star}(c_{i,x}) = \lim_{j \to \infty} \mathbf{\Pi}_i^{\star,(j)}(c_{i,x}) \ ,$$

and

$$\vec{V}_i^{\star}(x) = \lim_{j \to \infty} \vec{V}_i^{\star,(j)}(x) \ .$$

It follows the conclusion. ♣

We now proceed with the main proof. Let $x = 0$ and we obtain

$$\vec{V}_i^{\star}(0) = \mathbf{\Pi}_i^{\star}(c_{i,0}) \cdot \mathbf{F}_i \vec{\hat{V}}_i(0) \ .$$

We can also write the above relation for $x = 0$ as:

$$\begin{aligned}
\left(\frac{\vec{V}_i(0)}{\vec{V}_i'(0)}\right) &= \mathbf{\Pi}_i^{\star}(\Delta c_i) \left(\mathbf{F}_i' \middle| \mathbf{B}_i'\right) \left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\begin{array}{c|c} \mathbf{\Pi}_i(\Delta c_i) & \bar{\mathbf{\Pi}}_i^{\star}(\Delta c_i) \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \left(\begin{array}{c|c} \mathbf{F}_i & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\begin{array}{c|c} \mathbf{\Pi}_i(\Delta c_i)\mathbf{F}_i & \bar{\mathbf{\Pi}}_i^{\star}(\Delta c_i) \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right) \left(\frac{\vec{U}_{i+1}(0)}{\vec{U}_0(0)}\right) \\
&= \left(\frac{\mathbf{\Pi}_i(\Delta c_i)\mathbf{F}_i \vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}_i^{\star}(\Delta c_i)\vec{U}_0(0)}{\vec{U}_0(0)}\right).
\end{aligned}$$

As a result we can represent $\vec{V}_i(0)$ in the following matrix form

$$\vec{V}_i(0) = \mathbf{\Pi}_i(\Delta c_i)\mathbf{F}_i \vec{U}_{i+1}(0) + \bar{\mathbf{\Pi}}_i^{a}(\Delta c_i)\vec{U}_0(0) \ ,$$

by noting that $\mathbf{\Pi}_i$ is formed by the first k_i rows and columns of matrix $\mathbf{\Pi}_i^{\star}$ and $\bar{\mathbf{\Pi}}_i^{\star}$ is formed by the first k_i rows and the last $k_i^{\star} - k_i = k_0$ columns of $\mathbf{\Pi}_i^{\star}$. (8) follows from *Claim 1.*

For the case $i = m$, i.e., the last graph $\mathcal{P}_m$, the region size is infinite, therefore delay transitions do not exist. Recall that

$$\vec{U}_m(x) = \max_{\alpha \in Act} \left\{ \int_0^{\infty} \widehat{\mathbf{M}}_m^{\alpha}(\tau)\vec{U}_m(x+\tau)d\tau + \vec{1}_F + \int_0^{\infty} \mathbf{B}_m^{\alpha}(\tau)d\tau \cdot \vec{U}_0(0) \right\} \ .$$

We first prove the following claim:

Claim 3. For any $x \in \mathbb{R}_{\geq 0}$, $\vec{U}_m(x)$ is a constant vector function.

Proof of Claim 3. We define

$$\begin{aligned}
\vec{U}_m^{(0)}(x) &= \vec{0} \\
\vec{U}_m^{(j+1)}(x) &= \max_{\alpha \in Act} \left\{ \int_0^{\infty} \widehat{\mathbf{M}}_m^{\alpha}(\tau)\vec{U}_m^{(j)}(x+\tau)d\tau + \vec{1}_F + \int_0^{\infty} \mathbf{B}_m^{\alpha}(\tau)d\tau \cdot \vec{U}_0(0) \right\} \ .
\end{aligned}$$

It is not difficult to see that $\vec{U}_m(x) = \lim_{j\to\infty} \vec{U}_m^{(j)}(x)$. We shall show, by induction on j, that $\vec{U}_m^{(j)}(x)$ is a constant vector function.

- Base case. $\vec{U}_m^{(0)}(x) = \vec{0}$, which is clearly constant.
- Induction Hypothesis. $\vec{U}_m^{(j)}(x)$ is a constant vector function.

– Induction step.

$$\begin{aligned}
&\vec{U}_m^{(j+1)}(x) \\
&= \max_{\alpha \in Act} \left\{ \int_0^\infty \widehat{\mathbf{M}}_m^a(\tau) \vec{U}_m^{(j)}(x+\tau) d\tau + \vec{1}_F + \int_0^\infty \mathbf{B}_m^\alpha(\tau) d\tau \cdot \vec{U}_0(0) \right\} \\
&\overset{\text{I.H.}}{=} \max_{\alpha \in Act} \left\{ \int_0^\infty \widehat{\mathbf{M}}_m^a(\tau) \cdot \vec{U}_m^{(j)}(x) d\tau + \vec{1}_F + \int_0^\infty \mathbf{B}_m^\alpha(\tau) d\tau \cdot \vec{U}_0(0) \right\} \\
&= \max_{\alpha \in Act} \left\{ \int_0^\infty \widehat{\mathbf{M}}_m^a(\tau) d\tau \cdot \vec{U}_m^{(j)}(x) + \vec{1}_F + \int_0^\infty \mathbf{B}_m^\alpha(\tau) d\tau \cdot \vec{U}_0(0) \right\} .
\end{aligned}$$

The conclusion follows trivially. ♣

Since $\vec{U}_m(x)$ is constant vector function, we have that

$$\vec{U}_m(x) = \max_{\alpha \in Act} \left\{ \int_0^\infty \widehat{\mathbf{M}}_m^\alpha(\tau) d\tau \cdot \vec{U}_m(x) + \vec{1}_F + \int_0^\infty \mathbf{B}_m^\alpha(\tau) d\tau \cdot \vec{U}_0(0) \right\} .$$

Moreover, it is easy to see that $\int_0^\infty \widehat{\mathbf{M}}_m^\alpha(\tau) d\tau$ boils down to $\widehat{\mathbf{P}}_m^\alpha$ and $\int_0^\infty \mathbf{B}_m^a(\tau) d\tau$ boils down to $\widehat{\mathbf{B}}_m^\alpha$. Also we add the vector $\vec{1}_F$ to ensure that the probability to start from a state in G_F is one. Hence, (9) follows trivially. □