Time-Bounded Verification of CTMCs against MTL specifications ⋆ ⋆⋆

Taolue Chen, Marco Diciolla, Marta Kwiatkowska, and Alexandru Mereacre

Department of Computer Science, Oxford University,
Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom

Abstract. In this paper we study time-bounded verification of a finite continuous-time Markov chain (CTMC) $\mathcal{C}$ against a real-time specification, provided as a metric temporal logic (MTL) property φ. The key question is: what is the probability of the set of timed paths of $\mathcal{C}$ that satisfy φ over a time interval of fixed, bounded length? We provide approximation algorithms to solve these problems. We first derive a bound N such that timed paths of $\mathcal{C}$ with at most N discrete jumps are sufficient to approximate the desired probability up to ε. Then, for each discrete (untimed) path σ of length at most N, we generate timed constraints over variables determining the residence time of each state along σ, depending on the real-time specification under consideration. The probability of the set of timed paths, determined by the discrete path and the associated timed constraints, can thus be formulated as a multidimensional integral. Summing up all such probabilities yields the result.

1 Introduction

Verification of *continuous-time Markov chains* (CTMCs) has received much attention in recent years [5]. Thanks to considerable improvements of algorithms, (symbolic) data structures and abstraction techniques, CTMC model checking has emerged as a valuable analysis technique. Aided by powerful software tools, it has been adopted by researchers from, e.g., systems biology, queuing networks and dependability. To mention just a few practical applications, these models have been used to quantify the throughput of production lines, to determine the mean time between failure in safety-critical systems, and to identify bottlenecks in high-speed communication networks.

The focus of CTMC model checking [4] has primarily been on checking stochastic versions of the *branching-time* temporal logic CTL, such as *continuous stochastic logic* CSL [4]. The verification of *linear temporal logic* (LTL) properties reduces to applying well-known algorithms [14,10] to embedded discrete-time Markov chains (DTMCs). Linear-time properties equipped with timing constraints have only recently been considered. In particular, [7,8] treat linear *real-time* specifications that are given as *deterministic timed automata* (DTA). These 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?". Such properties

⋆ This work is supported by the ERC Advanced Grant VERIWARE.

⋆⋆ A longer version of this paper appeared in the proceedings of FORMATS11 [9]

Andrew V. Jones (Ed.): Proceedings of the 1st Imperial College Student Workshop (ICCSW '11),
pp. 17–23. September 29th–30th 2011, London UK.

cannot be expressed in CSL nor in its dialects [3,11]. Model checking DTA properties can be done by a reduction to computing the reachability probability in a *piecewise deterministic Markov process*, based on the product construction between the CTMC and DTA [8,6]. It remains a challenge to tackle more general real-time specifications like *Metric Temporal Logics* ([1,12], MTL).

For this reason, we study the time-bounded verification problem of a CTMC $\mathcal{C}$, against a real-time specification provided as an MTL formula φ. The key question is: what is the probability of the set of timed paths of $\mathcal{C}$ that satisfy φ over a fixed time interval $[0, T]$ where $T \in \mathbb{R}_{>0}$? We provide approximation algorithms to solve these problems. Given any $\varepsilon > 0$ a priori, we first derive a bound N such that it is sufficient only to consider timed paths of $\mathcal{C}$ with at most N discrete jumps to approximate the desired probability up to ε. Then, for each *discrete* (untimed) path σ of $\mathcal{C}$ of length at most N, we generate a family of linear constraints, $\mathcal{S}$, over variables determining the residence time of each state in σ. The discrete path σ, together with the associated timing constraints $\mathcal{S}$, determines a set of *timed* paths of $\mathcal{C}$, each of which satisfies φ. The probability of this set of timed paths can be formulated as a multidimensional integral, which can be calculated by Laplace transforms. Summing up all such probabilities yields the desired result. We believe these results are of independent interest, as they have potential usage in domains such as runtime verification.

The reader should notice that even though MTL is generally undecidable [2] (if we include singular intervals), this does not affect our algorithm. In fact, informally we can state that in any CTMC $\mathcal{C}$, the probability of an event happening in a specific singular time instant is zero.

2 Preliminaries

2.1 Continuous-time Markov chains

Given a set $\mathcal{H}$, let Pr: $\mathcal{F}(\mathcal{H}) \rightarrow [0, 1]$ be a *probability measure* on the measurable space $(\mathcal{H}, \mathcal{F}(\mathcal{H}))$, where $\mathcal{F}(\mathcal{H})$ is a σ-algebra over $\mathcal{H}$. Let $Distr(\mathcal{H})$ denote the set of probability measures on this measurable space.

Definition 1 (CTMC). *A (labeled) continuous-time Markov chain* (CTMC) *is a tuple* $\mathcal{C} = (S, \mathrm{AP}, L, \alpha, \mathbf{P}, E)$ *where*

- S *is a finite set of states;*
- AP *is a finite set of atomic propositions;*
- $L : S \rightarrow 2^{\mathrm{AP}}$ *is the labeling function;*
- $\alpha \in Distr(S)$ *is the initial distribution;*
- $\mathbf{P} : S \times S \rightarrow [0, 1]$ *is a stochastic matrix; and*
- $E : S \rightarrow \mathbb{R}_{\geq 0}$ *is the exit rate function.*

Example 1. An example CTMC is illustrated in Fig. 1, where $\mathrm{AP} = \{a, b, c\}$ and s_0 is the initial state, i.e., $\alpha(s_0) = 1$ and $\alpha(s) = 0$ for any $s \neq s_0$. The exit rates are indicated at the states, whereas the transition probabilities are attached to the transitions.

In a CTMC $\mathcal{C}$, state residence times are *exponentially* distributed. More precisely, the residence time X of a state $s \in S$ is a random variable governed by a nonnegative

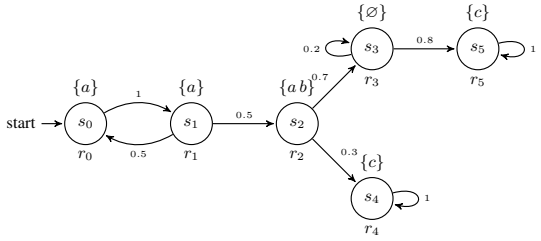

Fig. 1. An example CTMC

exponential distribution with parameter $E(s)$ (written as $X \sim \text{Exp}(E(s))$). Hence, the probability to exit state s in t time units (t.u. for short) is given by $\int_0^t E(s) \cdot e^{-E(s)\tau} d\tau$. Furthermore, the probability to take the transition from s to s' in t t.u. equals $\mathbf{P}(s, s') \cdot \int_0^t E(s) \cdot e^{-E(s)\tau} d\tau$.

Definition 2. *Given a* CTMC $\mathcal{C} = (S, \text{AP}, L, \alpha, \mathbf{P}, E)$*, we define the following notions.*

- *A (finite)* discrete *path $\sigma = s_0 \to s_1 \to s_2 \to \ldots$ is a (finite) sequence of states; we define σ_i to be the state s_i, and σ^i to be the prefix of length i of σ.*
- *A (finite)* timed *path $\rho = s_0 \xrightarrow{x_0} s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} \ldots$, where $x_i \in \mathbb{R}_{>0}$ for each $i \geq 0$, is a sequence starting in state s_0; we define $|\rho|$ to be the length of a* finite *timed path ρ; $\rho[n] := s_n$ is the n-th state of ρ and $\rho\langle n \rangle := x_n$ is the time spent in state s_n; let $\rho@t$ be the state occupied in ρ at time $t \in \mathbb{R}_{\geq 0}$, i.e. $\rho@t := \rho[n]$, where n is the smallest index such that $\sum_{i=0}^{n} \rho\langle i \rangle \geq t$.*

Intuitively, a timed path $\rho = s_0 \xrightarrow{x_0} s_1 \xrightarrow{x_1} s_2 \xrightarrow{x_2} \ldots$ suggests that the CTMC $\mathcal{C}$ starts in state s_0 and stays in this state for x_0 t.u., and then jumps to state s_1, staying there for x_1 t.u., and then jumps to s_2 and so on. An example timed path is $\rho = s_0 \xrightarrow{3} s_1 \xrightarrow{2} s_0 \xrightarrow{1.5} s_1 \xrightarrow{3.4} s_2 \ldots$ with $\rho[2] = s_0$ and $\rho@4 = \rho[1] = s_1$.

Let $Paths^{\mathcal{C}}$ denote the set of infinite timed paths in the CTMC $\mathcal{C}$, and $Paths^{\mathcal{C}}(s)$ the set of infinite timed paths in $\mathcal{C}$ that start in s. Given a time bound $T \in \mathbb{R}_{\geq 0}$ and $N \in \mathbb{N} \cup \{\infty\}$, we define $Paths^{\mathcal{C}}_{T,<N}(s)$, to be the set of all timed paths with *at most* $N - 1$ discrete jumps in time interval $[0, T]$; and $Paths^{\mathcal{C}}_{T,\geq N}(s)$, to be the set of all timed paths with *at least* N jumps in $[0, T]$.

For notational simplicity we will omit the superscript $\mathcal{C}$ when appropriate and also we write $Paths^{\mathcal{C}}_{T}$ instead of $Paths^{\mathcal{C}}_{T,\leq\infty}$ for the set of all timed paths with an arbitrary number of jumps in $[0, T]$.

In general, computing the probability of a cylinder set with k intervals $I_0 \ldots I_{k-1}$ (i.e. k discrete jumps) reduces to calculating k integrals over $I_0 \ldots I_{k-1}$.

2.2 Metric Temporal Logic

Definition 3 (Syntax of MTL). *Let* AP *be an arbitrary nonempty, finite set of atomic propositions. Let $I = [a, b]$ be an interval such that $a, b \in \mathbb{N} \cup \{\infty\}$. The* Metric

Temporal Logic *is inductively defined as:*

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \mathcal{U}_I \varphi_2 \ ,$$

where $p \in$ AP *and* φ_1, φ_2 *are* MTL *formulas.*

We introduce the time-bounded semantics for MTL, as follows.

Definition 4 (Semantics of MTL). *Given an* MTL *formula* φ*, a time bound* T*, a timed path* ρ *and a variable* $t \in \mathbb{R}_{\geq 0}$*, the satisfaction relation* $(\rho, t) \models_T \varphi$ *is inductively defined as follows:*

$$\begin{array}{ll}
(\rho,t) \models_T p & \Leftrightarrow p \in L(\rho@t) \wedge t \leq T \\
(\rho,t) \models_T \neg\varphi_1 & \Leftrightarrow (\rho,t) \not\models_T \varphi_1 \\
(\rho,t) \models_T \varphi_1 \wedge \varphi_2 & \Leftrightarrow (\rho,t) \models_T \varphi_1 \wedge (\rho,t) \models_T \varphi_2 \\
(\rho,t) \models_T \varphi_1 \mathcal{U}_I \varphi_2 & \Leftrightarrow \exists t'.\ t \leq t' \leq T \text{ s.t. } t' - t \in I \wedge (\rho,t') \models_T \varphi_2 \wedge \\
 & \quad\ \forall t''.\ t \leq t'' < t' \Rightarrow (\rho,t'') \models_T \varphi_1
\end{array}$$

where $p \in$ AP *and* φ_1, φ_2 *are* MTL *formulas.*

3 MTL Specifications

In this section we study the problem of model checking CTMCs against MTL properties. Let $\Pr^{\mathcal{C}}_T(\varphi) := \Pr^{\mathcal{C}}(\{\rho \in Paths^{\mathcal{C}}_T \mid (\rho, 0) \models_T \varphi\})$ denote the probability that the CTMC $\mathcal{C}$ satisfies the MTL formula φ, for a given time bound T. Instead of computing $\Pr^{\mathcal{C}}_T(\varphi)$, we give a procedure to compute $\Pr^{\mathcal{C}}_{T,<N}(\varphi) := \Pr^{\mathcal{C}}(Paths^{\mathcal{C}}_{T,<N}(\varphi))$ for sufficiently large N which ensures that $\Pr^{\mathcal{C}}_T(\varphi) - \Pr^{\mathcal{C}}_{T,<N}(\varphi) < \varepsilon$ for arbitrarily small $\varepsilon \in \mathbb{R}_{>0}$. This yields an approximation algorithm. Below we present an algorithm to compute $\Pr^{\mathcal{C}}_{T,<N}(\varphi)$. We first give a sketch, and provide the crucial sub-procedures in Sec. 3.1 and Sec. 3.2.

Choose N *to get the desired error bound* ε. The first step of the algorithm is to choose the smallest N such that we get the desired error bound ε.

Compute the product $\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}$. The basic idea of this step is to exclude those CTMC timed paths which definitely fail φ in order to reduce the number of paths to be analyzed. To this end, we define an LTL formula $\widetilde{\varphi}$ such that, if a discrete path of $\mathcal{C}$ fails $\widetilde{\varphi}$, then any timed path with the discrete path as the skeleton must fail φ. We then construct an NFA out of $\widetilde{\varphi}$ such that only those finite discrete CTMC paths which are accepted by the NFA are the prefixes of the potential skeletons of timed paths satisfying φ. Then we apply the standard product construction, which suffices to identify those CTMC finite discrete paths analyzed in the next step.

Any MTL formula φ can be transformed into a *positive normal form* containing only two temporal operators: $\mathcal{U}_{[a,b]}$ and $\Box_{[a,b]}$, where $(\rho, t) \models_T \Box_{[a,b]}\varphi$ iff $\forall t' \in [a, b] \Rightarrow (\rho, t + t') \models_T \varphi$.

Given any MTL φ *in positive normal form*, we define an (untimed) LTL formula $\widetilde{\varphi}$ as follows:

$$\begin{array}{ll}
\varphi = p & \Rightarrow \widetilde{\varphi} = p \\
\varphi = \neg p & \Rightarrow \widetilde{\varphi} = \neg p \\
\varphi = \varphi_1 \vee \varphi_2 & \Rightarrow \widetilde{\varphi} = \widetilde{\varphi}_1 \vee \widetilde{\varphi}_2 \\
\varphi = \varphi_1 \wedge \varphi_2 & \Rightarrow \widetilde{\varphi} = \widetilde{\varphi}_1 \wedge \widetilde{\varphi}_2 \\
\varphi = \varphi_1 \mathcal{U}_I \varphi_2 & \Rightarrow \widetilde{\varphi} = \widetilde{\varphi}_1 \mathcal{U} \widetilde{\varphi}_2 \\
\varphi = \Box_I \varphi_1 & \Rightarrow \widetilde{\varphi} = \text{TRUE}\, \mathcal{U} \widetilde{\varphi}_1
\end{array}$$

where φ_1 and φ_2 are MTL formulas and $\widetilde{\varphi}_1$ and $\widetilde{\varphi}_2$ are LTL formulas.

As the next step, we construct a *nondeterministic finite automaton* (NFA) $\mathcal{A}_{\widetilde{\varphi}}$ which accepts all the prefixes of infinite paths satisfying the formula $\widetilde{\varphi}$. The NFA can be obtained by a minor adaptation of the well-known Vardi-Wolper construction. We then build the product of $\mathcal{C}$ and $\mathcal{A}_{\widetilde{\varphi}}$ ($\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}$).

Compute all the discrete paths of $\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}$ of length at most N and calculate the probabilities.

1. Search the graph $\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}$ to get all the discrete accepting paths σ of $\mathcal{C}$ of length at most N;
2. Run Alg. 1 on each discrete path σ of length $n \leq N$ to obtain the system of linear inequalities $\mathcal{S}$;
3. Compute the probability of $\sigma[\mathcal{S}]$ (cf. Sec. 3.2);
4. Sum up all the probabilities for each discrete path to obtain $\Pr^{\mathcal{C}}_{T,<N}(\varphi)$.

3.1 Constraints Generation

We describe the Alg. 1 that takes as input a discrete path σ of length n and an MTL formula φ and returns a family of linear constraints $\mathcal{S} = \bigvee_{i \in I} \bigwedge_{j \in J_i} c_{ij}$ where c_{ij} is a linear inequality over the set of variables $t_0, \ldots, t_{n-1}$.

Algorithm 1 Constraints generation

Require: A finite discrete path σ of length $n > 0$, an MTL formula φ and a time bound T
Ensure: Family of linear inequalities $\mathcal{S}$ over $t_0, \ldots, t_{n-1}$

$\mathcal{S}'$:=`Constr_Gen(`σ`,0,`φ`)`
$\mathcal{S}$:=`Fourier_Motzkin(`$\mathcal{S}'$`,`t_0`,...,`t_{n-1}`)`
return $\mathcal{S}$

Function `Constr_Gen(`σ`,`t`,`φ`)`
case(φ) :

$\varphi = p$	:	**return** $\left(\bigvee_{k=0}^{n} p \in L(\sigma_k) \wedge \sum_{i=0}^{k} t_i \geq t \wedge \sum_{i=0}^{k-1} t_i < t\right) \wedge t < T$
$\varphi = \neg\varphi_1$	:	$\mathcal{S}' :=$ `¬Constr_Gen(`σ`,`t`,`φ_1`)`
$\varphi = \varphi_1 \wedge \varphi_2$	:	$\mathcal{S}' :=$`Constr_Gen(`σ`,`t`,`φ_1`)` $\wedge$ `Constr_Gen(`σ`,`t`,`φ_2`)`
$\varphi = \varphi_1 \mathcal{U}_{[a,b]} \varphi_2$	:	$\mathcal{S}' := \exists t'.\big(t \leq t' < T \wedge t' - t \geq a \wedge t' - t < b \wedge$ `Constr_Gen(`σ`,`t'`,`φ_2`)` $\wedge\, \forall t''.\, t \leq t'' < t' \Rightarrow$ `Constr_Gen(`σ`,`t''`,`φ_1`)`$\big)$

return $\mathcal{S}'$

3.2 Computing Probabilities

Given a CTMC $\mathcal{C}$, a discrete path σ of length N and the family of linear constraints $\mathcal{S}(t_0, \ldots, t_{N-1})$ obtained from Alg. 1, the main task of this section is to compute the probability of $\sigma[\mathcal{S}]$, i.e., $\Pr^{\mathcal{C}}(\sigma[\mathcal{S}])$. The value of the joint probability can be computed through the following multidimensional integration:

$$\Pr^{\mathcal{C}}(\sigma[\mathcal{S}]) = \underbrace{\int \cdots \int}_{N}{}_{\mathcal{S}(t_0,\ldots,t_{N-1})} \prod_{i=0}^{N-1} E(s_i) \cdot \mathbf{P}(s_i, s_{i+1}) \times e^{-E(s_i)\tau_i} d\tau_i. \quad (1)$$

We use the algorithm of [13] (Sec. 5) to compute efficiently the multidimensional integral based on the Laplace transform.

3.3 Main Algorithm

We summarize the time-bounded verification algorithm for a CTMC $\mathcal{C}$ against an MTL formula φ in Alg. 2. Recall that Λ is the maximal exit rate appearing in $\mathcal{C}$.

Algorithm 2 Time-bounded verification of a CTMC $\mathcal{C}$ against an MTL formula φ

Require: $\mathcal{C}$, φ, T and ε
Ensure: $\Pr^{\mathcal{C}}_{T,<N}(\varphi)$
  Choose an integer $N \geq \Lambda T e^2 + \ln(\frac{1}{\varepsilon})$
  Transform φ into $\widetilde{\varphi}$ and generate NFA $\mathcal{A}_{\widetilde{\varphi}}$ out of $\widetilde{\varphi}$
  Compute the product $\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}$
  for each discrete path σ of $(\mathcal{C} \otimes \mathcal{A}_{\widetilde{\varphi}}) \downarrow_1$ of length $n < N$ **do**
    Generate the family of linear constraints $\mathcal{S}(t_0, \ldots, t_{n-1})$
    Calculate the probability p of $\sigma[\mathcal{S}]$
    $\Pr^{\mathcal{C}}_{T,<N}(\varphi) := \Pr^{\mathcal{C}}_{T,<N}(\varphi) + p$
  end for
  return $\Pr^{\mathcal{C}}_{T,<N}(\varphi)$

4 Conclusion

In this paper we have studied time-bounded verification of CTMCs against real-time specifications. In particular, we presented effective procedures to approximate the probability of the set of timed paths of CTMCs that satisfy real-time specifications over a time interval of fixed bounded length, arbitrarily closely. Model checking CTMCs against linear real-time specifications has received scant attention so far. To our knowledge, this issue has only been (partially) addressed in [7,3,11].

A natural question is how to tackle the traditional (time-unbounded) verification. The scheme introduced in this paper still works. However, one cannot guarantee an approximation to stay within the given error bound ε, which means that the resulting procedure is *not* an approximation algorithm any more. We leave this as future work.

References

1. R. Alur and T. A. Henzinger. Real-time logics: Complexity and expressiveness. In *LICS*, pages 390–401, 1990.
2. R. Alur, T. Feder, and T. A. Henzinger. The benefits of relaxing punctuality. *J. ACM*, 43(1):116–146, 1996.
3. C. Baier, L. Cloth, B. R. Haverkort, M. Kuntz, and M. Siegle. Model checking Markov chains with actions and state labels. *IEEE Trans. Software Eng.*, 33(4):209–224, 2007.
4. C. Baier, B. R. Haverkort, H. Hermanns, and J.-P. Katoen. Model-checking algorithms for continuous-time Markov chains. *IEEE Trans. Software Eng.*, 29(6):524–541, 2003.
5. C. Baier, B. R. Haverkort, H. Hermanns, and J.-P. Katoen. Performance evaluation and model checking join forces. *Commun. ACM*, 53(9):76–85, 2010.
6. B. Barbot, T. Chen, T. Han, J.-P. Katoen, and A. Mereacre. Efficient CTMC model checking of linear real-time objectives. In P. A. Abdulla and K. R. M. Leino, editors, *TACAS*, volume 6605 of *Lecture Notes in Computer Science*, pages 128–142. Springer, 2011.
7. T. Chen, T. Han, J.-P. Katoen, and A. Mereacre. Quantitative model checking of continuous-time Markov chains against timed automata specifications. In *LICS*, pages 309–318, 2009.
8. T. Chen, T. Han, J.-P. Katoen, and A. Mereacre. Model checking of continuous-time Markov chains against timed automata specifications. *Logical Methods in Computer Science*, 7(1–2):1–34, 2011.
9. T. Chen, M. Diciolla, M. Kwiatkowska, and A. Mereacre. Time-Bounded Verification of CTMCs against Real-Time Specifications. 9th International Conference on Formal Modeling and Analysis of Timed Systems, FORMATS 2011, Aalborg, Denmark. *Lecture Notes in Computer Science*, 6919, 2011.
10. C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *J. ACM)*, 42(4):857–907, 1995.
11. S. Donatelli, S. Haddad, and J. Sproston. Model checking timed and stochastic properties with CSL^{TA}. *IEEE Trans. Software Eng.*, 35(2):224–240, 2009.
12. R. Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Systems*, 2(4):255–299, 1990.
13. J. B. Lasserre and E. S. Zeron. A Laplace transform algorithm for the volume of a convex polytope. *J. ACM*, 48(6):1126–1140, 2001.
14. M. Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *FOCS*, pages 327–338, 1985.