On Model-Checking Probabilistic Timed Automata against Probabilistic Duration Properties

Dang Van Hung[1], Miaomiao Zhang[2], Pham Dinh Chinh[1]

[1] University of Engineering and Technology, VNU, Hanoi, Vietnam [2]School of Software Engineering, Tongji University, Shanghai, China

Contact Information:
Information Technology
University of Engineering and Technology, Vietnam
National University, Hanoi
144 Xuan Thuy, Cau Giay, Hanoi, Vietnam

Email: `pdchinh@gmail.com`

Abstract

In this paper, we consider a subclass of Probabilistic Duration Calculus formula called Simple Probabilistic Duration Calculus (SPDC) as a language for specifying dependability requirements for real-time systems, and address the two problems: to decide if a probabilistic timed automaton satisfies a SPDC formula, and to decide if there is a strategy to choose an execution of a given automaton that satisfies a SPDC formula. We prove that the both problems are decidable for a class of SPDC called probabilistic linear duration invariants, and provide a model checking algorithm for solving these problems.

Introduction

In this paper, we introduced a simple probabilistic extension of DC called Probabilistic Duration Calculus for specifying dependability requirements of real-time systems. We use the behavioral model proposed by Kwiatkowska et al to define the semantics of our logic. Since probabilistic timed CTL and PDC are not comparable, and since for many probabilistic properties PDC is more convenient to specify, a model checking technique for checking probabilistic timed automata against PDC properties is useful. To solve this problem, we first develop a technique to decide if a strategy in a probabilistic timed automaton satisfies a PDC formula of a certain form. Then, we generalize this technique to achieve our goal with a model-checking algorithm.

The first version of this paper was published in [2]. In this extended version, in addition to the problem of verification, we formulate also the problem of strategy synthesis, i.e. to decide if there is a strategy for a probabilistic timed automaton that satisfies a probabilistic linear duration invariant and show that this problem is also solvable. We provide all proof details and algorithms for doing model-check.

Main Objectives

1. present the Probabilistic Timed Automata model.
2. presents syntax and semantics of our PDC.
3. presented in Section 4 where we formulate our model checking problem and give our solution to it.

Materials and Methods

In 1992, Chaochen Zhou, Hoare C.A.R and Anders Ravn introduced Duration Calculus [1] as a logic for reasoning about real-time systems. A version with a proof system of Probabilistic Duration Calculus with infinite interval was then developed by Dimitar Guelev [3], and in [4] we have shown that the calculus is useful for reasoning about QoS contracts in component-based real-time systems.

For Duration Calculus, some techniques for checking if a timed automaton satisfies a duration calculus formula written in the form of linear duration invariants have been developed. However, to our knowledge, not many works have been done for checking if a probabilistic real-time system satisfies a PDC formula.

Kwiatkowska et al in [5] proposed a variant of probabilistic timed automata that allows probabilistic choice only at discrete transitions.

Results

Probabilistic Timed Automata

Definition 1. *A probabilistic timed automaton (PTA) is a tuple $G = (\mathcal{S}, \mathcal{L}, \bar{s}, \mathcal{C}, inv, prob, \langle \tau_s \rangle_{s \in \mathcal{S}})$ consisting of*

- *a finite set $\mathcal{S}$ of nodes, a start node $\bar{s} \in \mathcal{S}$, a finite set $\mathcal{C}$ of clocks,*
- *a function $\mathcal{L} : \mathcal{S} \to 2^{AP}$ assigning to each node of the automaton a set of atomic propositions that are supposed to be those that are true in that node, a function $inv : \mathcal{S} \to \mathbf{Z}_{\mathcal{C}}$ assigning to each node an invariant condition,*
- *a function $prob : \mathcal{S} \to 2^{\mu(\mathcal{S} \times 2^{\mathcal{C}})}$ assigning to each node a set of discrete probability distributions on $\mathcal{S} \times 2^{\mathcal{C}}$,*
- *a family of functions $\langle \tau_s \rangle_{s \in \mathcal{S}}$ where, for any $s \in \mathcal{S}$, $\tau_s : prob(s) \to \mathbf{Z}_{\mathcal{C}}$ assigns to each $p \in prob(s)$ an enabling condition.*

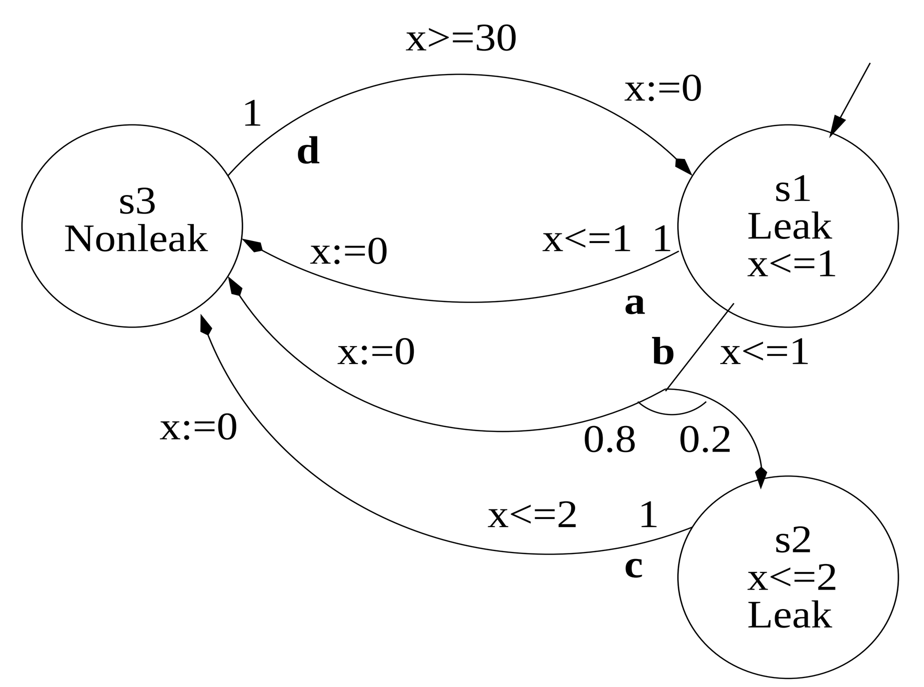

Figure 1: A probabilistic timed automaton for a simple gas burner

Definition 2. *A strategy (or scheduler) of a probabilistic timed structure $\mathcal{M} = (\mathcal{Q}, Steps, L)$ is a function A mapping every nonempty finite path ω of $\mathcal{M}$ to a pair (t, p) such that $A(\omega) \in Steps(last(\omega))$, and the empty path ϵ to a state in $\mathcal{Q}$. Let $\mathcal{A}$ be the set of all strategies of $\mathcal{M}$.*

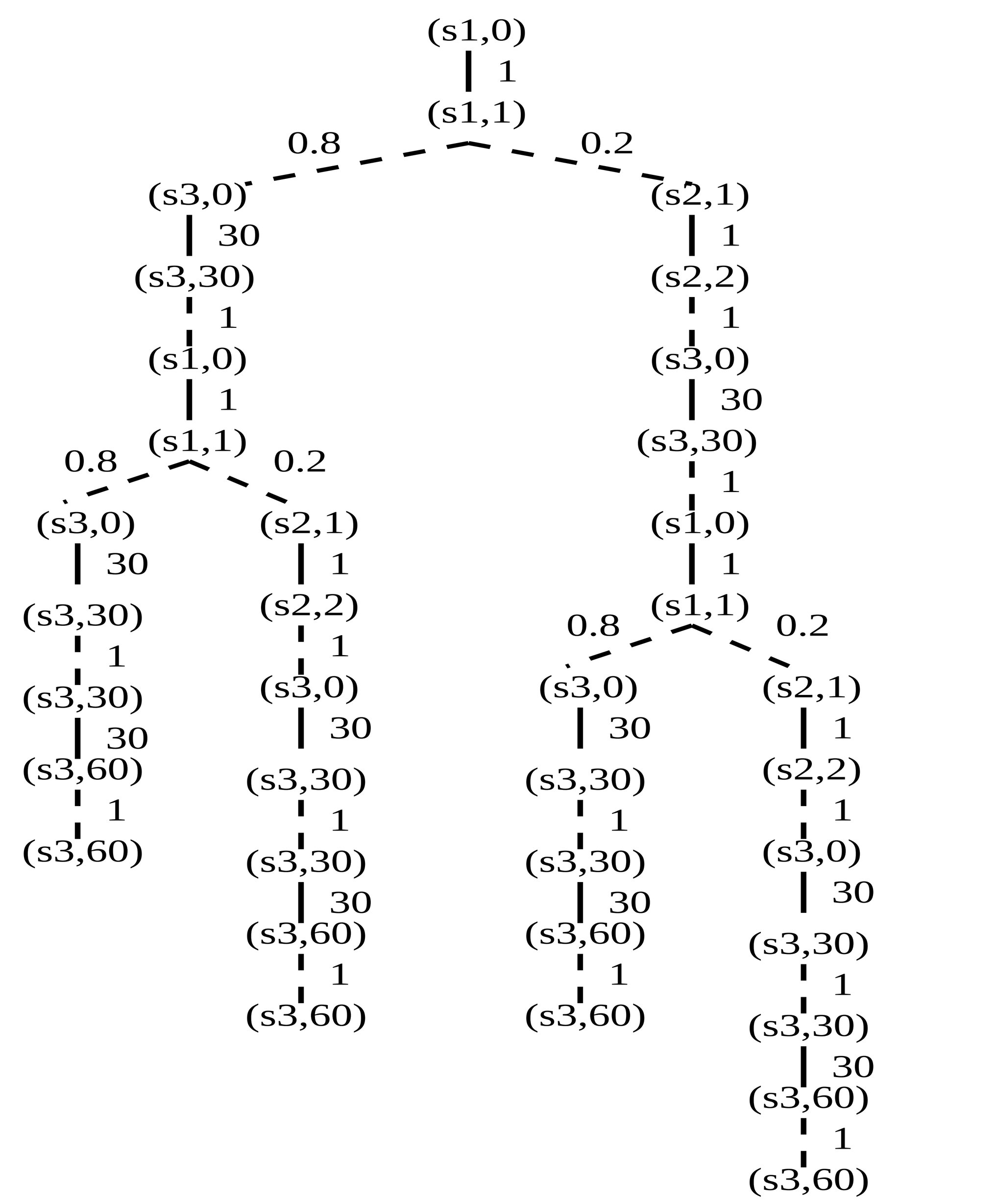

Figure 2: A part of a strategy A for the simple gas burner

Probabilistic Duration Calculus

In this section we introduce a simple form of Probabilistic Duration Calculus.

Definition 3. *Let R stand for relations (e.g. $\leq, =$), and F stand for functions (e.g. $+, -$). The syntax of Probabilistic Duration Calculus is defined as follows.*

$$\begin{aligned}
\Phi &::= \Psi \mid [\Psi]_{\exists\lambda} \mid \neg\Phi \mid \Phi \wedge \Phi, \\
\Psi &::= R(\eta, \ldots, \eta) \mid \neg\Psi \mid \Psi \wedge \Psi \mid \Psi; \Psi, \\
\eta &::= \textstyle\int S \mid F(\eta, \ldots, \eta), \\
S &::= \mathbf{1} \mid P \mid \neg S \mid S \wedge S,
\end{aligned}$$

where Φ stands for Probabilistic Duration Calculus formulas, Ψ stands for Duration Calculus formulas, η stands for duration terms, S stands for state expressions, and P is a symbol in the set of atomic proposition AP.

Model checking probabilistic timed automata against PDC properties

We are interested specially in the PDC formulas of the form $[\Psi]_{\exists\lambda}$, where Ψ has the form $\Box(a \leq \ell \leq b \Rightarrow \sum_{i=1}^{k} c_i \int P_i \leq M)$ called linear duration invariants (LDI), where M, a and b are integers, b could be ∞.

Now consider the following case for PDC formula Φ:

$$\Phi = [\Psi]_{\exists\lambda}, \quad \Psi = \Box\Psi1 \tag{1}$$

Theorem 1. *For a PDC formula Φ of the form (1) where Ψ is a linear duration invariant, it is decidable whether a finitely representable integral strategy A of probabilistic timed automaton G satisfies Φ at any time point t.*

Theorem 2. *For a PDC formula Φ of the form (1) where Ψ is a linear duration invariant, it is decidable whether Φ is satisfied by all integral strategies of a probabilistic timed automaton G at any time point.*

Theorem 3. *Given a PTA G and a PDC formula $\Phi = [\Psi]_{\exists\lambda}$, where Ψ is an LDI, we can decide if there exists a finitely representable strategy A such that $A, t \models_{PDC} [\Psi]_{\exists\lambda}$ for all t, and in the case such a strategy exists, we can find it.*

Conclusions

This paper has presented the problem of checking probabilistic timed automata against probabilistic duration calculus formulas. The problem is decidable for a class of PDC formulas of the form $[\Psi]_{\exists\lambda}$ where Ψ is a linear duration invariant, or a DC formula for bounded liveness. The technique for model checking is an extension of our techniques for checking if a timed automaton satisfies a linear duration invariant using a searching method in the integral region graph of the timed automaton. The complexity of the decision procedure is high in general. Since the problem possesses a potential high complexity, we have not implemented the technique yet. Hope that with the increasing computing power in the future, we can develop an effective tool for model-checking based on the technique.

Forthcoming Research

We are looking for some special cases of the problem which are simpler and still useful for which our technique can work well, and then implement it as a tool to assist checking the dependability for embedded systems.

References

[1] Z. Chaochen, C. A. R. Hoare, and A. P. Ravn, "A calculus of durations," *Information processing letters*, vol. 40, no. 5, pp. 269–276, 1991.

[2] D. Van Hung and M. Zhang, "On verification of probabilistic timed automata against probabilistic duration properties," in *null*, pp. 165–172, IEEE, 2007.

[3] D. P. Guelev, "Probabilistic interval temporal logic and duration calculus with infinite intervals: Complete proof systems," *arXiv preprint arXiv:0706.0692*, 2007.

[4] D. P. Guelev and D. Van Hung, "Reasoning about qos contracts in the probabilistic duration calculus," *Electronic Notes in Theoretical Computer Science*, vol. 238, no. 6, pp. 41–62, 2010.

[5] M. Kwiatkowska and D. Parker, "Automated verification and strategy synthesis for probabilistic systems," in *Automated Technology for Verification and Analysis*, pp. 5–22, Springer, 2013.

Acknowledgements

This research was funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.03-2014.23.