

# UBA-specifications are not that easy for Markov chains

## 1 Introduction

The standard probabilistic model-checking approach for finite-state Markovian models and formulas of linear temporal logic (LTL) relies on the analysis of the product of the Markovian model and a deterministic  $\omega$ -automaton representing the given LTL formula. While the quantitative analysis of the product can be carried out in time polynomial in the size of the model and the automaton, the translation of LTL formulas into deterministic automata can cause a double-exponential blow-up. From a complexity-theoretic point of view, this is optimal for Markovian models with nondeterminism [CY95]. However, the analysis of purely probabilistic models (Markov chains) against LTL formulas can be carried out in (single) exponential time using different methods, such as an iterative automata-less approach [CY95], separated Büchi automata rather than deterministic automata [CSS03] or (powerset constructions of) weak alternating automata [BRV04].

A further approach using unambiguous automata has been proposed by Benedikt, Lenhardt and Worrell [BLW13, Len13]. They first present a technique for computing the probability of a Markov chain to satisfy a (co-)safety specification given by an unambiguous finite automaton (UFA) using a linear equation system with variables for pairs of states in the Markov chain and the UFA. This approach can be seen as an elegant variant of the universality test for UFA using difference equations [SH85]. Furthermore, [BLW13, Len13] presents an algorithm for the quantitative analysis of Markov chains against  $\omega$ -regular properties specified by unambiguous Büchi automata (UBA). As observed by the authors, this approach is flawed. An attempt to repair the proposed UBA-based analysis of Markov chains has been presented by the authors in the arXiv document [BLW14]. However, the approach of [BLW14] is flawed again. This document serves to provide an example illustrating the faultiness of the algorithms for the analysis of Markov chains against UBA-specifications presented in [BLW14] (see Section 3) and of the algorithm presented in [BLW13], which is the same as in [Len13] (see Section 4).

## 2 Notations

An nondeterministic Büchi automaton is a tuple  $\mathcal{A} = (Q, Q_0, \Sigma, \delta, F)$  where  $Q$  is a finite set of states,  $Q_0 \subseteq Q$  is a set of initial states,  $\Sigma$  denotes the alphabet,  $\delta : Q \times \Sigma \rightarrow 2^Q$  denotes the transition

function, and  $F$  is a set of accepting states. We extend the transition function  $\delta : Q \times \Sigma \rightarrow 2^Q$  in the standard way for finite words.

Occasionally, for  $p, q \in Q$  and  $x \in \Sigma^*$ , we also write  $p \xrightarrow{x} q$  to indicate that  $q \in \delta(p, x)$ . A run for an infinite word  $w = a_1 a_2 a_3 \dots \in \Sigma^\omega$  is an infinite sequence  $\rho = q_0 q_1 \dots \in Q^\omega$  such that  $q_{i+1} \in \delta(q_i, a_{i+1})$  for all  $i \in \mathbb{N}$  and  $q_0 \in Q_0$ . Run  $\rho$  is called accepting, if  $q_i \in F$  for infinitely many  $i \in \mathbb{N}$ . The language  $\mathcal{L}_\omega(\mathcal{A})$  of accepted words consists of all infinite words  $w \in \Sigma^\omega$  that have at least one accepting run.  $\mathcal{A}$  is called *unambiguous* if each accepted word  $w \in \mathcal{L}_\omega(\mathcal{A})$  has exactly one accepting run. We use the shortform notations NBA and UBA for nondeterministic and unambiguous Büchi automata, respectively.

A Markov chain is a triple  $\mathcal{M} = (S, P, \iota)$  where  $S$  is a finite set of states,  $P : S \times S \rightarrow [0, 1]$  is the transition probability function satisfying  $\sum_{s' \in S} P(s, s') = 1$  for all states  $s \in S$  and  $\iota : S \rightarrow [0, 1]$  an initial distribution on  $S$ , i.e.,  $\sum_{s' \in S} \iota(s') = 1$  for all states  $s \in S$ . We write  $\Pr^\mathcal{M}$  to denote the standard probability measure for measurable sets of infinite paths of  $\mathcal{M}$ . If  $L \subseteq S^\omega$  is measurable then  $\Pr^\mathcal{M}(L)$  denotes the probability for  $\mathcal{M}$  to generate an infinite path  $\pi$  with  $\pi \in L$ . If  $K \subseteq S^*$  then  $\Pr^\mathcal{M}(K)$  is a shortform notation for  $\Pr^\mathcal{M}(L_K)$  where  $L_K$  is the set of infinite sequences  $w \in S^\omega$  that have a prefix in  $K$ . If  $\mu : S \rightarrow [0, 1]$  is a distribution then we write  $\Pr_\mu^\mathcal{M}$  for  $\Pr^{\mathcal{M}_\mu}$  where  $\mathcal{M}_\mu$  is the Markov chain  $(S, P, \mu)$ . This notation will be used for the distributions  $P(s, \cdot)$  where  $s$  is a state in  $\mathcal{M}$ . Thus,

$$\Pr_{P(s, \cdot)}^\mathcal{M} = \sum_{t \in S} P(s, t) \cdot \Pr_t^\mathcal{M} \quad \text{where} \quad \Pr_t^\mathcal{M} = \Pr_{\text{Dirac}[t]}^\mathcal{M}$$

and  $\text{Dirac}(t) : S \rightarrow [0, 1]$  denotes the Dirac distribution for state  $t$ , i.e.,  $\text{Dirac}[t] : S \rightarrow [0, 1]$  is given by  $\text{Dirac}[t](t) = 1$  and  $\text{Dirac}[t](s') = 0$  for  $s' \in S \setminus \{t\}$ .<sup>1</sup>

### 3 Counterexample for the [BLW14]-approach

In what follows, let  $\mathcal{M} = (S, P, \iota)$  be a Markov chain and  $\mathcal{U} = (Q, \Sigma, \delta, Q_0, F)$  an UBA with the alphabet  $\Sigma = S$ . The task addressed in [BLW14] is to compute  $\Pr^\mathcal{M}(\mathcal{L}_\omega(\mathcal{U}))$ .

The algorithm proposed in [BLW14] relies on the mistaken belief that if the Markov chain  $\mathcal{M}$  generates words accepted by the given UBA  $\mathcal{U}$  with positive probability then the product-graph  $\mathcal{M} \otimes \mathcal{U}$  contains *recurrent pairs*. These are pairs  $\langle s, q \rangle$  consisting of a state  $s$  in  $\mathcal{M}$  and a state  $q$  of  $\mathcal{U}$  such that almost all paths in  $\mathcal{M}$  starting in a successor of  $s$  can be written as the infinite concatenation of cycles around  $s$  that have a run in  $\mathcal{U}$  starting and ending in  $q$ . (The formal definition of recurrent pairs will be given below.) This claim, however, is wrong as there exist UBA that continuously need a look-ahead for the paths starting in a fixed state of the Markov chain.

Before presenting a counterexample illustrating this phenomenon and the faultiness of [BLW14], we recall some notations of [BLW14]. Given a state  $s \in S$  of the Markov chain  $\mathcal{M}$  and a state

<sup>1</sup>We depart here from the notations of [BLW14] where the notation  $\Pr_{\mathcal{M}, t}$  has been used as a shortform for  $\Pr_{P(t, \cdot)}^\mathcal{M}$ .

$q \in Q$  of the UBA  $\mathcal{U}$ , the regular languages  $G_{s,q}, H_{s,q} \subseteq S^+$  are defined as follows:

$$\begin{aligned} G_{s,q} &= \{ s_1 s_2 \dots s_k \in S^+ : s_k = s \text{ and } p \xrightarrow{s_1 s_2 \dots s_k} q \text{ for some } p \in Q_0 \} \\ H_{s,q} &= \{ s_1 s_2 \dots s_k \in S^+ : s_k = s \text{ and } q \xrightarrow{s_1 s_2 \dots s_k} q \} \end{aligned}$$

A pair  $\langle s, q \rangle \in S \times F$  is called *recurrent* if  $\Pr_{P(s, \cdot)}^{\mathcal{M}}(H_{s,q}^\omega) = 1$ .

The accepted language of  $\mathcal{U}$  can then be written as:

$$\mathcal{L}_\omega(\mathcal{U}) = \bigcup_{(s,q) \in S \times F} G_{s,q} \cdot H_{s,q}^\omega$$

The idea of [BLW14] is reduce the task to compute  $\Pr^{\mathcal{M}}(\mathcal{L}_\omega(\mathcal{U}))$  to the task to compute the probability for  $\mathcal{M}$  to generate a finite word accepted by an UFA for the language given by the UFA resulting from the union of the regular languages  $G_{s,q}$  where  $\langle s, q \rangle$  is recurrent. To show the correctness of this approach, [BLW14] claims that for each pair  $\langle s, q \rangle \in S \times Q$ :

$$\Pr_{P(s, \cdot)}^{\mathcal{M}}(H_{s,q}^\omega) \in \{0, 1\}$$

and thus  $\Pr_{P(s, \cdot)}^{\mathcal{M}}(H_{s,q}^\omega) = 0$  for the non-recurrent pairs  $\langle s, q \rangle \in S \times F$ . To prove this claim, the authors conjecture that:

$$\Pr_{P(s, \cdot)}^{\mathcal{M}}((H_{s,q})^\omega) = \lim_{n \rightarrow \infty} \Pr_{P(s, \cdot)}^{\mathcal{M}}((H_{s,q})^n) \stackrel{(*)}{=} \lim_{n \rightarrow \infty} \Pr_{P(s, \cdot)}^{\mathcal{M}}(H_{s,q})^n$$

The following example shows that equality  $(*)$  is wrong, and recurrent pairs need not to exist, even if  $\mathcal{U}$  is universal.

**Example.** We consider the Markov chain  $\mathcal{M} = (S, P, \iota)$  with two states, say  $S = \{a, b\}$ , and the transition probabilities

$$P(a, a) = P(a, b) = P(b, a) = P(b, b) = \frac{1}{2}$$

and the uniform initial distribution, i.e.,  $\iota(a) = \iota(b) = \frac{1}{2}$ . Thus:

$$\Pr_{P(a, \cdot)}^{\mathcal{M}} = \Pr_{P(b, \cdot)}^{\mathcal{M}} = \frac{1}{2} \cdot \Pr_a^{\mathcal{M}} + \frac{1}{2} \cdot \Pr_b^{\mathcal{M}}$$

From state  $a$ , the Markov chain  $\mathcal{M}$  schedules almost surely an infinite word  $w$  starting with  $a$  and containing both symbols  $a$  and  $b$  infinitely often. The analogous statement holds for state  $b$  of  $\mathcal{M}$ .

Let  $\mathcal{U} = (Q, \{a, b\}, \delta, Q, Q)$  be the UBA with state space  $Q = \{q_a, q_b\}$  where both states are initial and final and

$$\delta(q_a, a) = \delta(q_b, b) = \{q_a, q_b\}$$

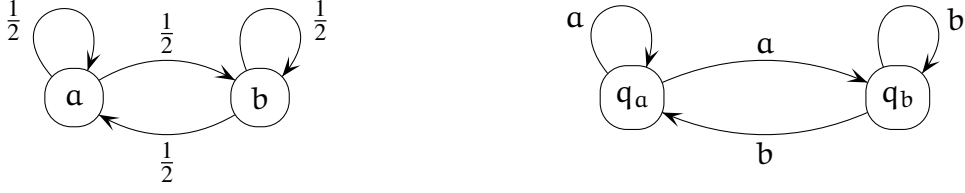


Figure 1: Markov chain  $\mathcal{M}$  (left) and universal UBA (right)

while  $\delta(q_a, b) = \delta(q_b, a) = \emptyset$ . Then,  $\mathcal{U}$  is universal as  $\mathcal{U}$  can use a one-letter look-ahead to generate an infinite run for each infinite word over  $\{a, b\}$ . More precisely, for doing so,  $\mathcal{U}$  moves to state  $q_a$  if the next letter is  $a$  and to state  $q_b$  if the next letter is  $b$ . As both states are final, each word has an accepting run. Thus,  $\mathcal{L}_\omega(\mathcal{U}) = \{a, b\}^\omega$  and therefore  $\Pr^\mathcal{M}(\mathcal{L}_\omega(\mathcal{U})) = 1$ .

The language  $H_{a, q_a}$  is given by the regular expression  $a(a + b)^*$ . Thus, for  $n \geq 2$ , the language  $H_{a, q_a}^n$  consists of all finite words  $x \in \{a, b\}^*$  that start with letter  $a$  and contain at least  $n$  occurrences of letter  $a$ . Likewise, the language  $H_{a, q_a}^\omega$  consists of all infinite words over  $\{a, b\}$  with infinitely many  $a$ 's and where the first letter is  $a$ . Hence:

$$\begin{aligned} \Pr_a^\mathcal{M}(H_{a, q_a}^\omega) &= \Pr_a^\mathcal{M}(H_{a, q_a}^n) = 1 \\ \Pr_b^\mathcal{M}(H_{a, q_a}^\omega) &= \Pr_b^\mathcal{M}(H_{a, q_a}^n) = 0 \end{aligned}$$

for all  $n \in \mathbb{N}$  with  $n \geq 1$ . This yields:

$$\Pr_{P(a, \cdot)}^\mathcal{M}(H_{a, q_a}^\omega) = \Pr_{P(a, \cdot)}^\mathcal{M}(H_{a, q_a}^n) = \frac{1}{2}$$

for all  $n \in \mathbb{N}$  with  $n \geq 1$ . On the other hand:

$$\lim_{n \rightarrow \infty} \Pr_{P(a, \cdot)}^\mathcal{M}(H_{a, q_a}^n) = \lim_{n \rightarrow \infty} \left(\frac{1}{2}\right)^n = 0$$

Thus, equality (\*) is wrong. In this example, none of the pairs  $\langle a, q_a \rangle$ ,  $\langle a, q_b \rangle$ ,  $\langle b, q_a \rangle$ ,  $\langle b, q_b \rangle$  is recurrent. Note that the languages  $H_{a, q_b}$  and  $H_{b, q_a}$  are empty and that an analogous calculation yields:

$$\Pr_{P(b, \cdot)}^\mathcal{M}(H_{b, q_b}^\omega) = \Pr_{P(b, \cdot)}^\mathcal{M}(H_{b, q_b}^n) = \frac{1}{2}$$

and  $\lim_{n \rightarrow \infty} \Pr_{P(b, \cdot)}^\mathcal{M}(H_{b, q_b}^n) = 0$ .

## 4 Counterexample for the [BLW13]-approach

A similar counterexample can be constructed for Lemma 7.1 of [BLW13] (p.22), i.e., the original proposal for using UBA for model checking of DTMCs.

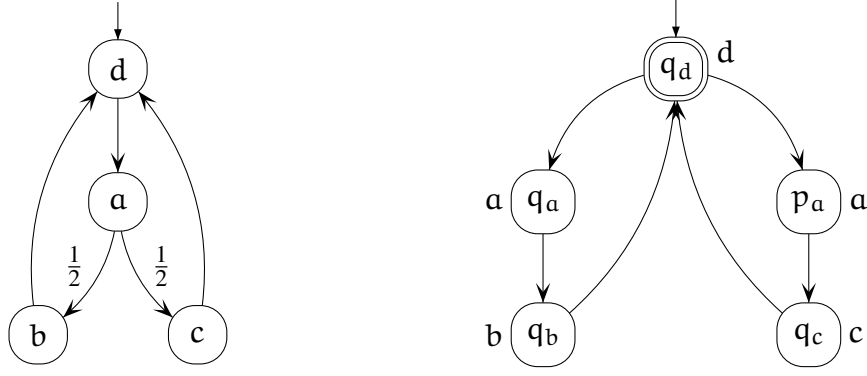


Figure 2: Markov chain  $\mathcal{M}$  (left) and UBA  $\mathcal{U}$  (right, state labels from alphabet  $\{a, b, c, d\}$ )

In [BLW13], the Büchi automata are state-labeled, usually with an alphabet over some atomic propositions. For presentational simplicity, we use here a fixed alphabet  $\{a, b, c, d\}$ , corresponding to the states of the Markov chain, omitting the atomic proposition based labeling functions. Consider the (state-labeled) UBA  $\mathcal{U}$  and the Markov Chain  $\mathcal{M}$  of Figure 2.

The UBA accepts all words of the form

$$((dab) + (dac))^\omega$$

and consequently  $\Pr^{\mathcal{M}}(\mathcal{L}(\mathcal{U})) = 1$ .

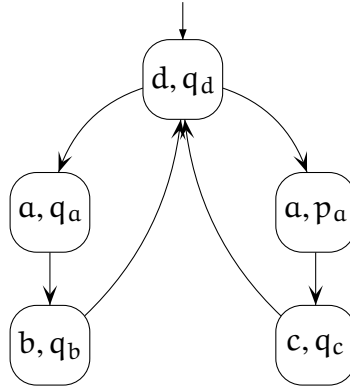


Figure 3: Product graph according to [BLW13]

The product graph  $\mathcal{M} \otimes \mathcal{U}$  that arises from the construction in the proof of Lemma 7.1 of [BLW13] is depicted in Figure 3.

The product graph is strongly connected, but it is not accepting as condition (ii) is violated: Consider the vertex  $(a, q_a)$  in the product graph. There exists a transition  $a \rightarrow c$  in the Markov chain,

however there is no successor  $(c, t)$  in the SCC of the product graph, with  $t$  being a successor of  $q_a$  in the UBA ( $c$  can not be consumed from the state  $q_a$  in the UBA). As the (only) SCC in the product is not accepting, all vertices in the product graph are assigned value 0 in the linear equation system, yielding that  $\Pr^{\mathcal{M}}(\mathcal{L}(\mathcal{U})) = 0$ . However, as stated above,  $\Pr^{\mathcal{M}}(\mathcal{L}(\mathcal{U})) = 1$ .

## References

- [BLW13] Michael Benedikt, Rastislav Lenhardt, and James Worrell. Two variable vs. linear temporal logic in model checking and games. *Logical Methods in Computer Science*, 9(2), 2013.
- [BLW14] Michael Benedikt, Rastislav Lenhardt, and James Worrell. Model checking Markov chains against unambiguous Büchi automata. *CoRR*, abs/1405.4560, 2014.
- [BRV04] Doron Bustan, Sasha Rubin, and Moshe Y. Vardi. Verifying  $\omega$ -regular properties of Markov chains. In *16th International Conference on Computer Aided Verification (CAV)*, volume 3114 of *Lecture Notes in Computer Science*, pages 189–201, 2004.
- [CSS03] Jean-Michel Couvreur, Nasser Saheb, and Grégoire Sutre. An optimal automata approach to LTL model checking of probabilistic systems. In *10th International Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR)*, volume 2850 of *Lecture Notes in Computer Science*, pages 361–375, 2003.
- [CY95] Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.
- [Len13] Rastislav Lenhardt. *Two Variable and Linear Temporal Logic in Model Checking and Games*. PhD thesis, University of Oxford, 2013.
- [SH85] Richard E. Stearns and Harry B. Hunt. On the equivalence and containment problem for unambiguous regular expressions, grammars, and automata. *SIAM Journal on Computing*, pages 598–611, 1985.