Probabilistic Temporal Logic for Reasoning about Bounded Policies

Nima Motamed, Natasha Alechina, Mehdi Dastani, Dragan Doder, Brian Logan

Utrecht University

Abstract

To build a theory of intention revision for agents operating in stochastic environments, we need a logic in which we can explicitly reason about their decision-making policies and those policies' uncertain outcomes. Towards this end, we propose PLBP, a novel probabilistic temporal logic for Markov Decision Processes that allows us to reason about finite traces and policies of bounded size. The logic is designed so that its expressive power is sufficient for the intended applications, whilst at the same time possessing strong computational properties. We prove that the satisfiability problem for our logic is decidable, and that its model checking problem is PSPACE-complete. This allows us to e.g. algorithmically verify whether an agents' intentions are coherent, or whether a specific policy satisfies safety and/or liveness properties.

To design intelligent and autonomous agents, it is important to develop a theory of intention revision, a topic receiving increased interest in recent decades. As observed by (Shoham 2009) and subsequent work, a primitive yet important kind of intention is that of commitment to perform an action at a specific time. This already comes with nontrivial complications, involving temporal reasoning about the pre- and postconditions of actions. To build a formal framework capable of dealing with this notion of intention, a natural approach is to start with an appropriate underlying temporal logic, as done by (Icard, Pacuit, and Shoham 2010; van Zee et al. 2020).

In order to build a theory of intention revision for agents operating in stochastic environments using behavioural policies, we require an appropriate probabilistic temporal logic on which to build the model. Such a logic should allow us to reason about the execution, precondition and (possibly many) postconditions of actions and policies. For practical purposes, the logic should possess strong computational properties such as efficient (or at least, decidable) model checking and satisfiability, so that we can e.g. algorithmically verify properties of policies and compute whether certain logical inferences are valid. Existing probabilistic temporal logics in the literature do not satisfy these desiderata: they do not allow such reasoning, and they generally high complexity/undecidable model checking and satisfiability. Furthermore, all of these logics are interpreted over infinite traces, while finite traces are both sufficient and more natural for most applications in AI.

The contribution of this paper is the introduction of the Probabilistic Logic of Bounded Policies (PLBP), a novel probabilistic temporal logic fitting our desiderata, interpreted over finite traces and bounded-time policies in Markov Decision Processes. The logic allows for both reasoning about specific actions/policies as well as the existence of policies satisfying certain properties. The model checking problem for PLBP is PSPACE-complete, and the satisfiability problem is decidable in 3EXPTIME. The novelty of the logic lies in the fact that it enables us to express a host of properties important for both general applications in AI, as well as for our intended applications, while simultaneously maintaining strong computational properties, which is uncommon amongst probabilistic temporal logics.

PLBP is defined relative to a countably infinite set Prop of *propositional variables* and a finite set $\mathscr{A}$ of *actions*. To each action $a \in \mathscr{A}$ we associate a *precondition* pre_a, which is a conjunction of literals over Prop, and a finite nonempty list Post_a of possible *postconditions*, which are also conjunctions of literals. We refer to the ith postcondition of a as $\mathsf{post}_{a,i}$. We require postconditions of an action a to be mutually inconsistent in the standard propositional sense.

Definition 1 (MDP) *A* Markov decision process *(MDP) over $\mathscr{A}$ is a tuple $\mathbb{M} = \langle S, P, V \rangle$, where S is the set of* states*, $P\colon S \times \mathscr{A} \rightharpoonup [0,1]^S$ is the* partial probabilistic transition function *with $P(s,a)$ a probability distribution (whenever defined), and $V\colon S \to 2^{\mathsf{Prop}}$ is the* valuation*.*

These are required to satisfy the following conditions. First, for all $s \in S$, there is some $a \in \mathscr{A}$ such that $s \models \mathsf{pre}_a$ in the standard propositional sense. Second, $P_{s,a}$ is defined iff $s \models \mathsf{pre}_a$. Third, given $P_{s,a}$ defined, (i) for all $t \in S$ such that $P_{s,a}(t) > 0$, there is a unique $\mathsf{post}_{a,i}$ in Post_a such that $t \models \mathsf{post}_{a,i}$, and (ii) for all $\mathsf{post}_{a,i}$ in Post_a there is a unique t such that $P_{s,a}(t) > 0$ and $t \models \mathsf{post}_{a,i}$.

The first condition states that MDPs have no deadlocks, and the other conditions ensure that pre- and postconditions are meaningful: an action is executable precisely when the precondition holds, and the possible outcomes of an action are precisely the postconditions.

Our logic will be built around the notion of an *n-step policy*, telling the agent how to act for n time steps in a memoryful manner. While such policies might seem restrictive compared to more standard notions of policy considered in

the literature, it suffices for our intended applications. And importantly, while n-step policies are still memoryful (in a bounded sense), the amount of n-step policies for some number of steps from a certain state in a finite MDP will always be finite, in contrast to general memoryful policies. This is an important property that is used in in proving the decidability of the satisfiability problem.

Definition 2 (n-step policies) *Given an MDP $\mathbb{M}$ and $n \geqslant 0$, an* n-step policy *is a pair $\langle s, \pi \rangle$ where $s \in S$ is referred to as its* initial state*, and $\pi \colon S_s^{\leqslant n} \to \mathscr{A}$ is a function such that $s_k \models \mathsf{pre}_{\pi(s_1 \cdots s_k)}$ for all $s_1 \cdots s_k \in S_s^{\leqslant n}$. By slight abuse of notation we usually write π^s instead of the pair $\langle s, \pi \rangle$ or the function π in order to make the initial state explicit.*

For an n-step policy π^s, we define the set $\mathsf{Paths}(\pi^s)$ as

$$\{s_1 a_1 \cdots s_n a_n s_{n+1} \mid s_1 = s \text{ and } \pi^s(s_1 s_2 \cdots s_i) = a_i \text{ and } P(s_i, a_i)(s_{i+1}) > 0 \text{ for all } 1 \leqslant i \leqslant n\}.$$

Given an n-step policy π^s, its path distribution *is the probability distribution $\mu_\pi^{\mathbb{M},s}$ over $\mathsf{Paths}(\pi^s)$ defined as $\mu_\pi^{\mathbb{M},s}(s_1 a_1 \cdots s_n a_n s_{n+1}) = \prod_{i=1}^n P(s_i, a_i)(s_{i+1})$. This extends to sets of paths in the standard way by summing. Whenever it is clear from the context, we drop the $\mathbb{M}$ from the superscript.*

Definition 3 (Syntax & semantics) *The* language of PBLP *is inductively defined by the grammar*

$$\begin{aligned} \varphi &::= \bot \mid x \mid \varphi \wedge \varphi \mid \neg\varphi \mid \Diamond^n_{\bowtie r} \Phi^{n+1}, \\ \Phi^1 &::= \varphi \\ \Phi^{n+1} &::= \varphi \mid \mathsf{do}_a \mid \Phi^{n+1} \wedge \Phi^{n+1} \mid \neg\Phi^{n+1} \mid \mathsf{X}\Phi^n, \end{aligned}$$

where $x \in \mathsf{Prop}$, $a \in \mathscr{A}$, $n \geqslant 1$, $r \in \mathbb{Q} \cap [0,1]$, and $\bowtie \in \{<, =, >\}$. Formulas φ are referred to as state formulas*, and formulas Φ^n are referred to as* n-path formulas *(or more generally,* path formulas*).*

Given an MDP $\mathbb{M}$, the semantics of PBLP *is defined via simultaneous induction over state and path formulas. Propositional variables, Booleans and $\bot$ are interpreted as standard, so we only specify the semantics of the other formulas. For state formulas, we have for states s in $\mathbb{M}$ that $s \models \Diamond^n_{\bowtie r} \Phi$ iff there exists an n-step policy π^s such that $\mu_\pi^s(\{\mathbf{s} \in \mathsf{Paths}(\pi^s) \mid \mathbf{s} \models \Phi\}) \bowtie r$. For path formulas, we have for a path $\mathbf{s} = s_1 a_1 \cdots a_{n-1} s_n$ that $\mathbf{s} \models \varphi \iff s_1 \models \varphi$, $\mathbf{s} \models \mathsf{do}_a \iff a_1 = a$, and $\mathbf{s} \models \mathsf{X}\Phi \iff \mathbf{s}_{\mathsf{X}} \models \Phi$, where $\mathbf{s}_{\mathsf{X}} = s_2 a_2 \cdots s_n$.*

The modal formula $\Diamond^n_{\bowtie r} \Phi$ states that "the agent can act in the next n steps in such a way that Φ will hold with probability $\bowtie r$," do_a states that "the agent will now execute a," and X stands for the next-time operator. Note that we can define a universally quantified modality $\Box^n_{\bowtie r}$ as an abbreviation in the standard way.

General examples of what one can express with this logic include e.g. $\mathsf{pre}_a \wedge \Box^1_{\geqslant 0.6}(\mathsf{do}_a \to \mathsf{X}\varphi)$ stating "the agent can do a, and doing so causes φ to hold afterwards with probability at least 0.6," and $\Diamond^1_{=1}\mathsf{X}\Box^1_{=1}\mathsf{X}\neg\varphi$, stating "the agent can act now such that it becomes guaranteed that no matter how she acts in the next step, φ will not hold." Of particular interest to our intended applications are the following two examples.

First, note that we can express properties of specific policies as well by making use of postconditions. E.g. consider a 2-step policy saying to do a now (with two postconditions), and afterwards b_1 if we got the first postcondition of a, otherwise b_2. We can express in PBLP that under this policy the agent will be in a state satisfying φ with probability 0.5 with the formula $\Diamond^2_{=0.5}(\mathsf{do}_a \wedge \bigwedge_{i=1,2} \mathsf{X}(\mathsf{post}_{a,i} \to \mathsf{do}_{b_i}) \wedge \mathsf{XX}\varphi)$. Second, given a finite set I of intentions as 'commitments of actions towards time', i.e. pairs $\langle a, t \rangle$ of actions and time steps, it is important to be able to determine whether adopting these intentions is coherent w.r.t. the agent's beliefs, in the sense laid out by (Shoham 2009). Representing the agent's beliefs as a set Γ of formulas, we can formulate coherence of I with respect to Γ through determining the satisfiability of Γ together with $\mathsf{exec}_\theta(I) = \Diamond^{t_{\max}}_{\geqslant\theta} \bigwedge_{\langle a,t \rangle \in I} \mathsf{X}^t \mathsf{do}_a$, where $t_{\max} = \max_{\langle a,t \rangle \in I} t$. The resulting notion of coherence satisfies (stochastic generalisations of) the desiderata listed by (Shoham 2009).

Our logic possesses decidable model checking and satisfiability. Model checking is the problem of determining for an MDP, state s and formula φ whether $s \models \varphi$. We have a nondeterministic polynomial-space algorithm deciding this problem, giving us inclusion in PSPACE (since NPSPACE = PSPACE). We also have PSPACE-hardness by a reduction from QSAT inspired by (Bulling and Jamroga 2010). However, instead of the strategic setting of their logic, where a verifier has a strategy enforcing the 'yes' state if and only if the formula is satisfiable, we work in a stochastic setting where the agent has a policy reaching the 'yes' state with probability 1 if and only if the QBF formula is satisfiable.

Theorem 1 *The model checking problem for PBLP is PSPACE-complete.*

The satisfiability problem is that of determining for an input formula φ whether there exists an MDP and s such that $s \models \varphi$. Noting that PBLP has the bounded model property by a standard unravelling argument, our algorithm for this problem proceeds by iterating over state sets S up to the required bound and then determining whether there is a P such that the resulting MDP satisfies φ by determining whether a certain existential first-order logic formula is true in the theory of real-closed fields. The boundedness of both traces and policies is crucial to the procedure.

Theorem 2 *The satisfiability problem for PBLP is decidable in 3EXPTIME.*

References

Bulling, N.; and Jamroga, W. 2010. Verifying agents with memory is harder than it seemed. *AI Commun.*, 23(4): 389–403.

Icard, T.; Pacuit, E.; and Shoham, Y. 2010. Joint Revision of Belief and Intention. In *Proceedings of KR'10*, 572–574.

Shoham, Y. 2009. Logical Theories of Intention and the Database Perspective. *J. Philos. Log.*, 38(6): 633–647.

van Zee, M.; Doder, D.; van der Torre, L.; Dastani, M.; Icard, T.; and Pacuit, E. 2020. Intention as commitment toward time. *Artif. Intell.*, 283: 103270.