Probabilistic Temporal Logic for Reasoning about Bounded Policies

Nima Motamed
Utrecht University

Work with Natasha Alechina, Mehdi Dastani,
Dragan Doder, and Brian Logan

AAAI-SSS On the Effectiveness of Temporal Logics on Finite Traces in AI
March 2023

Yet another temporal logic...

Our intention is to study **intentions**: to build a theory of intention revision under uncertainty

Yet another temporal logic...

Our intention is to study **intentions**: to build a theory of intention revision under uncertainty

Reasoning about such intentions requires an appropriate probabilistic temporal logic, allowing us to explicitly reason about the execution and executability of the agent's **actions**

But well-known (**infinite-trace**) ones like PCTL, pCTL*, PATL/PATL* or Probabilistic Strategy Logic do not have this! Plus: they have high complexity (sometimes undecidable) **model checking**, and the decidability of **satisfiability** is still open

PBLP

We introduce the **Probabilistic Logic of Bounded Policies (PBLP)**

PBLP

We introduce the **Probabilistic Logic of Bounded Policies (PBLP)**

The logic is interpreted w.r.t. **finite traces** and **bounded policies** - a policy/strategy that holds for a certain number of steps.

PBLP is expressive enough for our needs (and can in fact express properties important for other AI applications), and it has good computational properties.

Markov Decision Processes and bounded policies

Fix a finite set $\mathcal{A}$ of **actions**, and for every $a \in \mathcal{A}$ a **precondition** pre_a and finite **set of postconditions** Post_a - these are conjunctions of literals.

Definition

An **MDP** is a tuple $\mathbb{M} = \langle S, P, V \rangle$, where S is a set of *states*, $P : S \times \mathcal{A} \rightsquigarrow \Delta(S)$ is the *partial probabilistic transition function*, and $V : S \to 2^{\mathsf{Prop}}$ is the *valuation*.

(+ coherence conditions ensuring that pre- and postconditions are meaningful)

Markov Decision Processes and bounded policies

Fix a finite set $\mathcal{A}$ of **actions**, and for every $a \in \mathcal{A}$ a **precondition** pre_a and finite **set of postconditions** Post_a - these are conjunctions of literals.

Definition

An **MDP** is a tuple $\mathbb{M} = \langle S, P, V \rangle$, where S is a set of *states*, $P : S \times \mathcal{A} \rightsquigarrow \Delta(S)$ is the *partial probabilistic transition function*, and $V : S \to 2^{\mathsf{Prop}}$ is the *valuation*.

(+ coherence conditions ensuring that pre- and postconditions are meaningful)

Definition

For $n \geqslant 1$, an ***n*-step policy from a state *s*** is a function $\pi : S_s^{\leqslant n} \to \mathcal{A}$ (where $S_s^{\leqslant n}$ is the set of all length $\leqslant n$ sequences from s) such that $s_k \models \mathsf{pre}_{\pi(s_1 \cdots s_k)}$.

Syntax & semantics

n-step path formulas (defined inductively w.r.t. n):

$$\Phi^0 ::= \varphi \quad \Phi^{n+1} ::= \varphi \mid \mathrm{do}_a (a \in A) \mid \Phi^{n+1} \wedge \Phi^{n+1} \mid \neg \Phi^{n+1} \mid \mathrm{X}\Phi^n$$

Interpreted over **state-action paths** $\mathbf{w} = s_1 a_1 \cdots s_n a_n s_{n+1}$, with $\mathbf{w} \models \mathrm{do}_a$ iff $a_1 = a$, and $\mathbf{w} \models \mathrm{X}\Phi$ iff $s_2 a_2 \cdots s_{n+1} \models \Phi$

State formulas:

$$\varphi ::= x (\in \mathrm{Prop}) \mid \varphi \wedge \varphi \mid \neg \varphi \mid \Diamond^n_{\bowtie r} \Phi^n$$

here, $n \geqslant 1$, $r \in [0, 1]$ *and* $\bowtie \in \{<, =, >\}$

For states: $s \models \Diamond^n_{\bowtie r} \Phi$ iff there is an n-step policy π from s such that under the policy, the probability that the next n steps of states and actions satisfies Φ is $\bowtie r$

What can we express?

- $\mathrm{pre}_a \wedge \Box^{1}_{\geqslant 0.8}(\mathrm{do}_a \rightarrow \mathrm{X}\varphi)$ "The agent can execute *a*, and doing so will cause φ to hold afterwards with probability at least 80%"

What can we express?

- $\mathrm{pre}_a \wedge \Box^1_{\geqslant 0.8}(\mathrm{do}_a \rightarrow \mathsf{X}\varphi)$ "The agent can execute a, and doing so will cause φ to hold afterwards with probability at least 80%"
- We can reason about **specific policies**: consider a 2-step policy saying to do a now, and afterwards b_1 if we got the first postcondition of a, otherwise b_2. The formula $\Diamond^2_{=0.6}(\mathrm{do}_a \wedge \bigwedge_{i=1,2} \mathsf{X}(\mathrm{post}_{a,i} \rightarrow \mathrm{do}_{b_i}) \wedge \mathsf{XX}\varphi)$ states that under this policy, φ holds with probability 60% in two steps.

What can we express?

- $\mathrm{pre}_a \wedge \Box^1_{\geqslant 0.8}(\mathrm{do}_a \to \mathsf{X}\varphi)$ “The agent can execute a, and doing so will cause φ to hold afterwards with probability at least 80%”
- We can reason about **specific policies**: consider a 2-step policy saying to do a now, and afterwards b_1 if we got the first postcondition of a, otherwise b_2. The formula $\Diamond^2_{=0.6}(\mathrm{do}_a \wedge \bigwedge_{i=1,2} \mathsf{X}(\mathrm{post}_{a,i} \to \mathrm{do}_{b_i}) \wedge \mathsf{XX}\varphi)$ states that under this policy, φ holds with probability 60% in two steps.
- Following Shoham (2009) in considering basic intentions to be pairs (a, t) denoting ‘the agent intends to do a at time t’, **coherence** of a set I of such intentions with respect to a set Γ of formulas representing the agent’s beliefs is stating that

$$\Gamma \cup \{\Diamond^{t_{\max}}_{>0} \bigwedge_{(a,t)\in I} \mathsf{X}^t \mathrm{do}_a\} \quad (\text{where } t_{\max} = \max_{(a,t)\in I} t)$$

is **satisfiable** - the agent does not believe that their intentions are not realizable

Computational properties

Model checking is **PSPACE-complete**: membership is shown using an NPSPACE-algorithm that traverses the MDP by guessing actions to take; hardness is shown by a reduction from QSAT inspired by Bulling & Jamroga (2010)

Computational properties

Model checking is **PSPACE-complete**: membership is shown using an NPSPACE-algorithm that traverses the MDP by guessing actions to take; hardness is shown by a reduction from QSAT inspired by Bulling & Jamroga (2010)

More excitingly: satisfiability is decidable in **2-EXPSPACE**: PBLP has the bounded model property, so the algorithm iterates over S and V up to the bound, and for each determines whether there is P and s satisfying the formula by checking whether a certain existential first-order logic sentence is valid in the theory of real closed fields

Wrapping up

Summing up:

- We have a logic allowing us to reason about **finite traces** and **bounded policies**
- The logic can express important properties for AI, while still having **strong computational properties**

Wrapping up

Summing up:

- We have a logic allowing us to reason about **finite traces** and **bounded policies**
- The logic can express important properties for AI, while still having **strong computational properties**

Future work:

- Defining and/or axiomatizing **belief and intention revision operators** using PBLP
- Develop a **quantitative extension** incorporating reward signals, for applications in Reinforcement Learning

Wrapping up

Summing up:

- We have a logic allowing us to reason about **finite traces** and **bounded policies**
- The logic can express important properties for AI, while still having **strong computational properties**

Future work:

- Defining and/or axiomatizing **belief and intention revision operators** using PBLP
- Develop a **quantitative extension** incorporating reward signals, for applications in Reinforcement Learning

Thank you!