A Probabilistic Finite Temporal Logic for Policies and Intentions

Nima Motamed, Natasha Alechina, Mehdi Dastani, Dragan Doder

August 26, 2022 — LAMAS&SR 2022 — Rennes, France

Utrecht University

A Probabilistic Finite Temporal Logic for Policies and Intentions

Nima Motamed, Natasha Alechina, Mehdi Dastani, Dragan Doder

August 26, 2022 — LAMAS&SR 2022 — Rennes, France

Utrecht University

A Probabilistic Finite Temporal Logic for Policies and Intentions ???

Nima Motamed, Natasha Alechina, Mehdi Dastani, Dragan Doder

August 26, 2022 — LAMAS&SR 2022 — Rennes, France

Utrecht University

Motivation: intentions

Developing intention revision with uncertainty

Following common BDI approach by working within a temporal logic

Motivation: intentions

Developing intention revision with uncertainty

Following common BDI approach by working within a temporal logic

Could use probabilistic strategy logic, probabilistic CTL

But no (known) decidable satisfiability + model checking (in some cases)...

Motivation: intentions

Developing intention revision with uncertainty

Following common BDI approach by working within a temporal logic

Could use probabilistic strategy logic, probabilistic CTL

But no (known) decidable satisfiability + model checking (in some cases)...

So here comes our logic: Finite MDP-PCTL*

Actions and propositions

Finite set $\mathcal{A}$ of actions, countable set Prop of propositional variables

For each action $a \in \mathcal{A}$, a precondition $\mathsf{pre}_a \in \mathsf{Prop}$, and finite set of postconditions $\mathsf{Post}_a \subseteq \mathsf{Prop}$

Actions have uncertain outcomes, i.e. multiple possible postconditions

Markov Decision Processes

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*.

(here, $\Delta(S)$ is the set of probability distributions on S, and $\rightsquigarrow$ denotes a partial function)

Markov Decision Processes

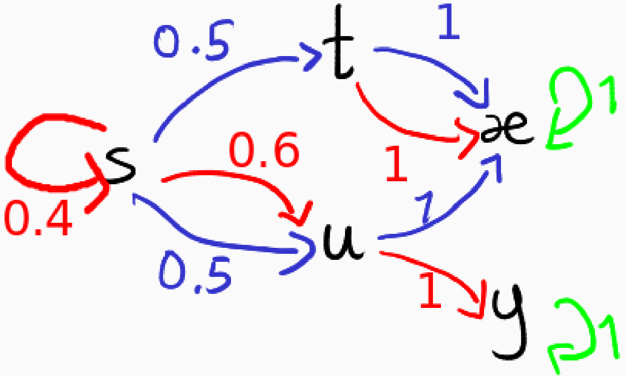
Definition

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*.

(here, $\Delta(S)$ is the set of probability distributions on S, and $\rightsquigarrow$ denotes a partial function)

MDPs are required to satisfy some coherence conditions, but these are not important here, except for that they entail $P(s, a)$ is finitely supported

Markov Decision Processes



Policies

How do we determine probability of reaching a state or sequence of states?

Policies!

Usually these are functions $S \to \mathcal{A}$ or $S \to \Delta(\mathcal{A})$, but in our setting we consider (deterministic) finite policies with (bounded) recall

Policies

Let S_s^n be the set of sequences in S of length n starting from s (similar definition for $S_s^{\leqslant n}$)

Definition

For $n \geqslant 0$, an n-step policy is a pair $\langle s, \pi \rangle$ of an *initial state* $s \in S$, and a function $\pi : S_s^{\leqslant n} \to \mathcal{A}$ such that always $\mathrm{pre}_{\pi(s_1 \cdots s_k)} \in V(s_k)$.

Note that we allow 0-step policies: here the agent has made no planning

Policies

Let S_s^n be the set of sequences in S of length n starting from s (similar definition for $S_s^{\leqslant n}$)

Definition

For $n \geqslant 0$, an n-step policy is a pair $\langle s, \pi \rangle$ of an *initial state* $s \in S$, and a function $\pi : S_s^{\leqslant n} \to \mathcal{A}$ such that always $\mathrm{pre}_{\pi(s_1 \cdots s_k)} \in V(s_k)$.

Note that we allow 0-step policies: here the agent has made no planning

Given an n-step policy $\langle \pi, s \rangle$, we get a probability distribution $\mu_s^\pi \in \Delta(S_s^{n+1})$ defined as

$$\mu_s^\pi(s_1 \cdots s_{n+1}) = \prod_{i=1}^{n} P(s_i, \pi(s_1 \cdots s_i))(s_{i+1})$$

Syntax

State formulas

$$\varphi ::= \bot \mid x \mid \mathrm{do}_a \mid \varphi \wedge \varphi \mid \neg\varphi \mid \Diamond^{n}_{\bowtie r} \Phi^{n+1}$$

n-step history formulas

$$\Phi^1 ::= \varphi \quad \Phi^{n+1} ::= \varphi \mid \Phi^{n+1} \wedge \Phi^{n+1} \mid \neg\Phi^{n+1} \mid \mathrm{X}\Phi^{n}$$

$x \in \mathrm{Prop}$, $a \in \mathcal{A}$, $n \geqslant 1$, $\bowtie \in \{<, \leqslant, =, \neq, \geqslant, >\}$, $r \in [0,1] \cap \mathbb{Q}$

Semantics

For finite policies $\langle s, \pi \rangle$

$$
\begin{aligned}
\pi, s &\models x \text{ iff } x \in V(s) \\
\pi, s &\models \mathrm{do}_a \text{ iff } \pi \text{ is} \geqslant 1\text{-step, and } \pi(s) = a \\
\pi, s &\models \Diamond^n_{\bowtie r}\Phi \text{ iff there exists an } n\text{-step policy } \rho \\
&\qquad \text{with initial state } s \text{ such that} \\
&\qquad \mu^\rho_s(\{\mathbf{s} \in S^{n+1}_s \mid \rho, \mathbf{s} \models \Phi\}) \bowtie r
\end{aligned}
$$

For n-step s, π and $\mathbf{s} \in S^{n+1}_s$

$$
\begin{aligned}
\pi, \mathbf{s} &\models \varphi \text{ iff } \pi, s \models \varphi \\
\pi, \mathbf{s} &\models \mathrm{X}\Phi \text{ iff } \pi^{s_2}, s_2 \cdots s_{n+1} \models \Phi
\end{aligned}
$$

where π^{s_2} is "π shifted forward to start from s_2"

Example formulas

$$\mathsf{pre}_a \land \Box^{1}_{\geqslant 0.8}(\mathsf{do}_a \to \mathsf{X}\varphi)$$

"The agent can execute a, and doing so will cause φ to hold afterwards with probability at least 80%"

Example formulas

$$\mathsf{pre}_a \wedge \Box^1_{\geqslant 0.8}(\mathsf{do}_a \rightarrow \mathsf{X}\varphi)$$

"The agent can execute *a*, and doing so will cause φ to hold afterwards with probability at least 80%"

$$\Box^n_{=1}\mathsf{X}^n\Diamond^m_{\geqslant 0.5}\mathsf{X}^m\varphi$$

"No matter how the agent acts in the next *n* steps, it is certain that he can afterwards make it at least 50% likely that φ holds in the next *m* steps"

Example formulas

Of interest to intention revision:

Consider intentions as commitments to perform an action at a certain time, i.e. intentions are pairs $\langle a, n \rangle$ with $a \in \mathcal{A}$ and $n \geqslant 1$. Given a set I of intentions and some $\theta \in [0, 1]$, the formula

$$\mathsf{exec}(I, \theta) = \Diamond_{\geqslant\theta}^{n_{\max}} \bigwedge_{\langle a,n \rangle \in I} \mathsf{X}^n \mathsf{do}_a$$

where $n_{\max} = \max_{\langle a,n \rangle \in I} n$ denotes that the agent can execute all his intentions with certainty $\geqslant \theta$

So coherence of intentions w.r.t. the agent's beliefs can be defined through satisfiability of the beliefs and exec

Decidability

Model checking is (obviously?) decidable - satisfiability is also decidable!

To see this, we observe that we can bound model size by the policy depth $\mathrm{pd}(\varphi)$ of a formula φ, denoting the deepest amount of steps the policies being quantified over in φ reach.

Theorem (Small model property)

If a formula φ is satisfiable, then it is satisfiable in an MDP with at most as many states as a tree with depth $\mathrm{pd}(\varphi)$ and branching factor $\sum_{a \in \mathcal{A}} |\mathrm{Post}_a|$, at a policy of at most $\mathrm{pd}(\varphi)$ steps.

(verified using standard unraveling techniques)

Decidability

Theorem (Satisfiability)

The problem of determining for a formula φ whether there exists an MDP and finite policy $\langle s, \pi \rangle$ such that $\pi, s \models \varphi$, is decidable.

Proof sketch. Using the bound z given by the small model property, iterate over sets $S = \{s_1, \ldots, s_y\}$ with $y \leqslant z$ and valuations V (need only consider those propositional variables appearing in φ plus pre- and postconditions)

Decidability

Theorem (Satisfiability)

The problem of determining for a formula φ whether there exists an MDP and finite policy $\langle s, \pi \rangle$ such that $\pi, s \models \varphi$, is decidable.

Proof sketch. Using the bound z given by the small model property, iterate over sets $S = \{s_1, \ldots, s_y\}$ with $y \leqslant z$ and valuations V (need only consider those propositional variables appearing in φ plus pre- and postconditions)

We construct a FOL sentence α in the language of real closed fields (RCF) such that

$$\alpha \text{ is true in the theory of RCF} \iff \text{there exists } P \text{ and } \langle s, \pi \rangle \text{ such that } \pi, s \models \varphi \text{ in } \langle S, P, V \rangle$$

Decidability

Note that we can encode P through variables $p_{i,j,a}$ denoting $P(s_i, a)(s_j)$ (taking care to allow $P(s_i, a)$ to be undefined), and that we can define an RCF formula $\beta(\mathbf{p})$ that is true in RCF iff the variables encode a well-defined P.

Decidability

Note that we can encode P through variables $p_{i,j,a}$ denoting $P(s_i, a)(s_j)$ (taking care to allow $P(s_i, a)$ to be undefined), and that we can define an RCF formula $\beta(\mathbf{p})$ that is true in RCF iff the variables encode a well-defined P.

If we can define an RCF formula $\gamma_{\varphi,i,\pi}(\mathbf{p})$ that is true in RCF together with $\beta(\mathbf{p})$ iff $\pi, s_i \models \varphi$, we could put

$$\alpha = \exists \mathbf{p}(\beta(\mathbf{p}) \wedge \bigvee_{\substack{n \leqslant \mathrm{pd}(\varphi) \\ n\text{-step } s_i, \pi}} \gamma_{\varphi,i,\pi}(\mathbf{p})),$$

and the well-known decidability of RCF would then give us our decision procedure.

Decidability

Get ready for a lot of subscripts. Leave now while you can!

We (simultaneously) define the following RCF formulas:

- $\delta_{n,i,X,\pi,\bowtie,r}$ (where $X \subseteq S_{s_i}^{n+1}$ and π is n-step), meaning "$\mu_{s_i}^{\pi}(X) \bowtie r$"

Decidability

Get ready for a lot of subscripts. Leave now while you can!

We (simultaneously) define the following RCF formulas:

- $\delta_{n,i,X,\pi,\bowtie,r}$ (where $X \subseteq S_{s_i}^{n+1}$ and π is n-step), meaning "$\mu_{s_i}^{\pi}(X) \bowtie r$"
- $\kappa_{n,\Phi,\mathbf{w},\pi}$ (where $\mathbf{w} \in S^{n+1}$ and π is n-step from w_1), meaning "$\pi, \mathbf{w} \models \Phi$"

  E.g. $\kappa_{n,\mathrm{X}\Phi,\mathbf{w},\pi} = \kappa_{n-1,\Phi,w_2\cdots w_{n+1},\pi^{w_2}}$

Decidability

Get ready for a lot of subscripts. Leave now while you can!

We (simultaneously) define the following RCF formulas:

- $\delta_{n,i,X,\pi,\bowtie,r}$ (where $X \subseteq S_{s_i}^{n+1}$ and π is n-step), meaning "$\mu_{s_i}^{\pi}(X) \bowtie r$"
- $\kappa_{n,\Phi,\mathbf{w},\pi}$ (where $\mathbf{w} \in S^{n+1}$ and π is n-step from w_1), meaning "$\pi, \mathbf{w} \models \Phi$"

  E.g. $\kappa_{n,\mathsf{X}\Phi,\mathbf{w},\pi} = \kappa_{n-1,\Phi,w_2 \cdots w_{n+1},\pi^{w_2}}$
- $\lambda_{\Phi,i,n,\pi,X}$ meaning "X is the set of all $\mathbf{w} \in S_{s_i}^{n+1}$ such that $\pi, s_i \models \Phi$"

  Defined as $\lambda_{\Phi,i,n,\pi,X} = \bigwedge_{\mathbf{w} \in X} \kappa_{n,\Phi,\mathbf{w},\pi} \wedge \bigwedge_{\mathbf{w} \in S_{s_i}^{n+1} \setminus X} \neg\kappa_{n,\Phi,\mathbf{w},\pi}$

Decidability

Get ready for a lot of subscripts. Leave now while you can!

We (simultaneously) define the following RCF formulas:

- $\delta_{n,i,X,\pi,\bowtie,r}$ (where $X \subseteq S_{s_i}^{n+1}$ and π is n-step), meaning "$\mu_{s_i}^{\pi}(X) \bowtie r$"
- $\kappa_{n,\Phi,\mathbf{w},\pi}$ (where $\mathbf{w} \in S^{n+1}$ and π is n-step from w_1), meaning "$\pi, \mathbf{w} \models \Phi$"
  E.g. $\kappa_{n,\mathrm{X}\Phi,\mathbf{w},\pi} = \kappa_{n-1,\Phi,w_2\cdots w_{n+1},\pi^{w_2}}$
- $\lambda_{\Phi,i,n,\pi,X}$ meaning "X is the set of all $\mathbf{w} \in S_{s_i}^{n+1}$ such that $\pi, s_i \models \Phi$"
  Defined as $\lambda_{\Phi,i,n,\pi,X} = \bigwedge_{\mathbf{w}\in X} \kappa_{n,\Phi,\mathbf{w},\pi} \wedge \bigwedge_{\mathbf{w}\in S_{s_i}^{n+1}\setminus X} \neg\kappa_{n,\Phi,\mathbf{w},\pi}$
- And the $\gamma_{\psi,i,\pi}$ mentioned before
  E.g. $\gamma_{\Diamond^n_{\bowtie r}\Phi,i,\pi} = \bigvee_{n\text{-step } s_i,\, \rho} \bigvee_{X\subseteq S_{s_i}^{n+1}} \lambda_{\Phi,i,n,\rho,X} \to \delta_{n,i,X,\pi,\bowtie,r}$

Take a breath now. We're having drinks in a bit.

Wrapping up

Summing up:

- We have a logic allowing us to reason about finite histories, MDPs, and policies
- The logic has decidability of both model checking & satisfiability

Wrapping up

Summing up:

- We have a logic allowing us to reason about finite histories, MDPs, and policies
- The logic has decidability of both model checking & satisfiability

Future work:

- Intention revision!
- Quantitative extensions, closer to AI/RL practice

Wrapping up

Summing up:

- We have a logic allowing us to reason about finite histories, MDPs, and policies
- The logic has decidability of both model checking & satisfiability

Future work:

- Intention revision!
- Quantitative extensions, closer to AI/RL practice
- *And if the heavens allow it, a better name for the logic*

nima@motamed.nl — n.motamed@uu.nl — https://motamed.nl

Wrapping up

Summing up:

- We have a logic allowing us to reason about finite histories, MDPs, and policies
- The logic has decidability of both model checking & satisfiability

Future work:

- Intention revision!
- Quantitative extensions, closer to AI/RL practice
- *And if the heavens allow it, a better name for the logic*

Thank you!

`nima@motamed.nl` — `n.motamed@uu.nl` — `https://motamed.nl`