Research Directions: Cyber-Physical Systems

www.cambridge.org/cbp

Causal temporal reasoning for Markov decision processes

Milad Kazemi, Jessica Lally and Nicola Paoletti

Department of Informatics, King's College London, London, UK

Abstract

We present *PCFTL (Probabilistic CounterFactual Temporal Logic)*, a new probabilistic temporal logic for the verification of Markov Decision Processes (MDP). PCFTL introduces operators for causal inference, allowing us to express interventional and counterfactual queries. Given a path formula ϕ, an interventional property is concerned with the satisfaction probability of ϕ if we apply a particular change I to the MDP (e.g., switching to a different policy); a counterfactual formula allows us to compute, given an observed MDP path τ, what the outcome of ϕ would have been had we applied I in the past and under the same random factors that led to observing τ. Our approach represents a departure from existing probabilistic temporal logics that do not support such counterfactual reasoning. From a syntactic viewpoint, we introduce a counterfactual operator that subsumes both interventional and counterfactual probabilities as well as the traditional probabilistic operator. This makes our logic strictly more expressive than PCTL*. The semantics of PCFTL rely on a structural causal model translation of the MDP, which provides a representation amenable to counterfactual inference. We evaluate PCFTL in the context of safe reinforcement learning using a benchmark of grid-world models.

Results

Cite this article: Kazemi M, Lally J, and Paoletti N (2025). Causal temporal reasoning for Markov decision processes. *Research Directions: Cyber-Physical Systems*. **3**, e3, 1–14. https://doi.org/10.1017/cbp.2025.2

Received: 2 May 2024
Revised: 23 January 2025
Accepted: 31 January 2025

Keywords:
causality; temporal logic; verification

Corresponding author: Milad Kazemi;
E-mail: milad.kazemi@kcl.ac.uk

1. Introduction

Temporal logic (TL) is arguably the primary language for the formal specification and reasoning about system correctness and safety. It has been successfully applied to the analysis of a wide range of systems, including cyber-physical systems (Bartocci et al., 2018), programs (Manna and Pnueli, 2012), and stochastic models (Kwiatkowska et al., 2007). In cyber-physical systems, TLs are especially useful for expressing and verifying critical properties of these systems, to ensure systems meet performance and safety criteria. For example, (probabilistic) TLs can express safety and reachability properties (e.g., *'will the system eventually reach the goal state(s) while avoiding unsafe states?"*) and fault-tolerance properties (e.g., *'will the system return to some desired service level after a fault?"*).

However, a limitation of existing TLs is that TL specifications must be evaluated on a fixed configuration of the system, for example a fixed choice of control policy, communication protocol, or system dynamics. That is, they cannot express queries like *'what is the probability that the system throughput will stay above a certain threshold if we switch to a high-performance controller?"*, or *'what would have been the probability that the signal would have stayed below a given threshold if we had used a different policy in the past?"* This kind of reasoning about different system conditions falls under the realm of causal inference (Pearl, 2009), by which the first query is called an *intervention* and the second a *counterfactual*. Even though both causal inference and TL-based verification are well-established on their own, their combination hasn't been sufficiently explored in past literature (see Section 7 for a more complete account of the related work). With this paper, we contribute to bridging these two fields.

We introduce *PCFTL (Probabilistic CounterFactual Temporal Logic)*, the first probabilistic temporal logic that explicitly includes causal operators to express interventional properties (*'what will happen if . . . "*), counterfactual properties (*'what would have happened if . . . "*), and so-called *causal effects*, defined as the difference of interventional or counterfactual probabilities between two different configurations. In particular, in this paper we focus on the analysis of *Markov Decision Processes (MDPs)*, which are capable of modeling sequential decision-making processes under uncertainty, a key aspect in many cyber-physical systems applications. MDPs provide a useful framework for a variety of applications, such as reinforcement learning, planning, and probabilistic verification. For MDPs, arguably the most relevant kind of causal reasoning concerns evaluating how a change in the MDP policy affects some outcome. The outcome of interest for us is the satisfaction probability of a temporal-logic formula.

Interventions are 'forward-looking" (Oberst and Sontag, 2019), as they allow us to evaluate the probability of a TL property ϕ after applying a particular change $X \leftarrow X'$ to the system. Counterfactuals are instead 'retrospective" (Oberst and Sontag, 2019), telling us what might have happened under a different condition: having observed an MDP path τ, they allow us to

© The Author(s), 2025. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

evaluate ϕ on the *what-if* version of τ, that is the path that we would have observed if we had applied $X \leftarrow X'$ at some point in the past, *provided that the random factors that yielded τ remain fixed.* Causal effects (Guo et al., 2020) allow us to establish the impact of a given change at the level of the individual path or overall, and they quantify the increase in the probability of ϕ induced by a manipulation $X \leftarrow X'$. Causal and counterfactual reasoning has gained a lot of attention in recent years due to its power in observational data studies: with counterfactuals, one can answer *what-if* questions relative to an observed path, that is without having to intervene on the real system (which might jeopardize safety) but using observational data only. Our PCFTL logic enables this kind of reasoning in the context of formal verification.

Our approach to incorporating causal inference in temporal logic involves only a minimal extension of traditional probabilistic logics. PCFTL is an extension of PCTL* (Baier et al., 1997; Baier, 1998) where the probabilistic operator $P_{\bowtie p}(\phi)$, which checks whether the probability of ϕ satisfies threshold $\bowtie p$ (where $\bowtie \in \{\leq, <, >, \geq\}$), is replaced with a counterfactual operator $I_{@t}.P_{\bowtie p}(\phi)$, which concerns the probability of ϕ if we had applied intervention I at t time steps in the past. Albeit minimal, such an extension provides great expressive power: if $t > 0$, then the operator corresponds to a counterfactual query; if $t = 0$, it represents an interventional probability; if both $t = 0$ and I is empty, then we retrieve the traditional $P_{\bowtie p}(\phi)$ operator.

Motivating example: To better grasp interventions and counterfactuals, consider an example of a robot in a 2D space, modeled by the equation $S_{t+1} = S_t + A_t + U_t$, where $S_t \in \mathbb{R}^2$ and $A_t \in \mathbb{R}^2$ are the state and action at time t, and $U_t \in \mathbb{R}^2$ is an *unobserved* random exogenous input (e.g., white Gaussian noise). The robot must satisfy a bounded safety property $\phi = \neg\mathcal{F}_{[1,4]}(S_t \geq [1, 2])$, which specifies that the robot must avoid entering the unsafe region $S_t \geq [1, 2]$ on all paths (up to length 3) that it takes. Suppose we observe a path τ under some policy π, given by $\tau = [0, 0] \xrightarrow{[0,1]} [0.1, 0.5] \xrightarrow{[1,1]} [0.8, 1.75] \xrightarrow{[0,0]} [1.3, 2.1]$, where $s \xrightarrow{a} s'$ denotes a step from state s to s' through action a. This path, and hence policy π, is unsafe because it violates the safety property in its final state. A question then arises: given τ, if we had intervened in the past by changing the policy from π to some π', could have we prevented this violation? Define the intervention $I = \pi \leftarrow \pi'$. Then, the counterfactual PCFTL query $I_{@3}.P_{\bowtie p}(\phi)$ allows us to evaluate the probability of the safety property ϕ in a what-if version of τ where we apply I (i.e., policy π' instead of π) at 3 steps back from the last state of τ, that is from the beginning of the path in this case.[1] In particular, the counterfactual path is obtained by applying I but by *keeping the same values of the random exogenous factors U_t that led to τ*. These factors cannot be directly observed, but, given the above Equation, they can be readily determined as $U_t = S_{t+1} - S_t - A_t$, leading to $U_1 = [0.1, -0.5]$, $U_2 = [-0.3, 0.25]$, and $U_3 = [0.5, 0.35]$. Then, suppose the alternative policy π' chooses actions $A'_1 = [0, 0.5]$ and $A'_3 = [-0.4, -0.2]$ (but keeps $A'_2 = A_2$), then this induces the counterfactual path $\tau' = [0, 0] \xrightarrow{[0,0.5]} [0.1, 0] \xrightarrow{[1,1]} [0.8, 1.25] \xrightarrow{[-0.4,-0.2]} [0.9, 1.4]$. Notably, now τ' satisfies the safety property.

Despite the simplicity of this example, counterfactual reasoning becomes challenging when dealing with discrete-state probabilistic models like MDPs. Indeed, the state of an MDP evolves according to a categorical distribution, for which the identification and inference of the exogenous factors are non-trivial.

Contributions: In this paper, we introduce the syntax and semantics of PCFTL and present a statistical model-checking approach for verifying PCFTL properties in MDP environments. Our approach, summarized in Figure 1, relies on translating the MDP into a so-called *structural causal model (SCM)*, a fundamental model in causal inference that enables computation of counterfactual distributions. We use a particular form of SCMs (Oberst and Sontag, 2019) suitable for encoding categorical counterfactuals (arising with discrete-state MDPs). After performing counterfactual inference, the SCM model is then translated back into an MDP amenable for PCFTL model checking. Unlike existing logics, PCFTL formulas are interpreted with respect to an observed MDP path τ, rather than a single MDP state, as we must keep track of the past to perform counterfactual reasoning.

Using efficient statistical model checking procedures, we evaluate PCFTL on a reinforcement learning benchmark (Chevalier-Boisvert et al., 2018) involving multiple 2D grid-world environments, goal-oriented tasks, and interventional and counterfactual properties under various policies learned through neural-network-based reinforcement learning methods. These results demonstrate the usefulness of PCFTL in AI safety, but our approach could enhance the verification of probabilistic models in a variety of domains, from distributed systems to security and biology.

The paper covers background about SCMs, MDPs, and SCM-based encoding of MDPs in Section 2, construction of counterfactual MDPs in Section 3, definition of PCFTL syntax, semantics, and decision procedures in Section 4, experimental results in Section 6, related work in Section 7, and conclusions in Section 8.

2. Background

2.1. Causal inference with structural causal models

Structural Causal Models (SCMs) (Pearl, 2009; Glymour et al., 2016) are equation-based models to specify and reason about causal relationships involving some variables of interest.

Definition 1. (Structural Causal Model (SCM)). *An SCM is a tuple $\mathcal{M} = (\mathbf{U}, \mathbf{V}, \mathcal{F}, P(\mathbf{U}))$ where*

- $\mathbf{U}$ *is a set of (mutually independent) exogenous variables.*
- $\mathbf{V}$ *is a set of endogenous variables, where the value of each $V \in \mathbf{V}$ is determined by a function $V = f_V(\mathbf{PA}_V, U_V)$. Here, $\mathbf{PA}_V \subseteq \mathbf{V}$ are the set of direct causes of V, and $U_V \in \mathbf{U}$.*
- $\mathcal{F}$ *is the set of functions $\{f_V\}_{V \in \mathbf{V}}$.*
- $P(\mathbf{U}) = \bigotimes_{U \in \mathbf{U}} P(U)$ *is the joint distribution of the (mutually independent) exogenous variables.*

Assignments in $\mathcal{F}$ must be acyclic, to ensure that no variable can be a direct or indirect cause of itself. Because of this, the causal relationships in an SCM can be represented by a directed acyclic graph (DAG), called a causal diagram.

In an SCM, the values of the exogenous variables $\mathbf{U}$ are determined by factors outside the model, which is modelled by some distribution $P(\mathbf{U})$. Exogenous variables are *unobserved* variables which act as the source of randomness in the system. Indeed, for a fixed realization u of $\mathbf{U}$, that is a concrete unfolding of the system's randomness, the values of $\mathbf{V}$ become deterministic, as they are uniquely determined by u and the causal processes $\mathcal{F}$. A concrete value u of $\mathbf{U}$ is also called *context* (or unit). We denote by

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

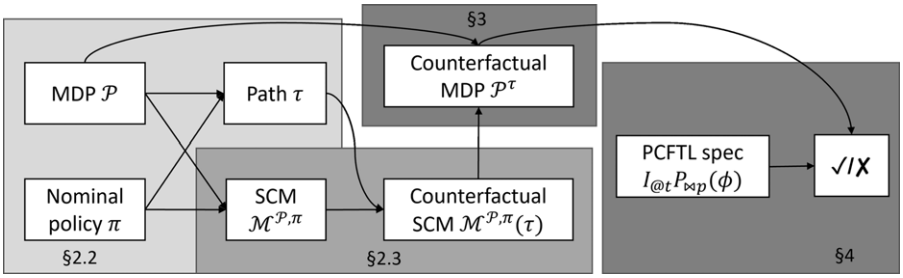


Figure 1. Overview of our approach to PCFTL verification, with section pointers.

$P_{\mathcal{M}}(\mathbf{V})$ the so-called *observational distribution* of $\mathbf{V}$, that is, the data-generating distribution entailed by the SCM $\mathcal{F}$ and $P(\mathbf{U})$.

Interventions. With SCMs, one can establish the causal effect of some input variable X on some output variable Y by evaluating Y after 'forcing" some specific values x on X, an operation called *intervention*. Applying $X \leftarrow x$ means replacing the RHS of $X = f_X(\mathbf{PA}_X, U_X)$ with x. Interventions allow to establish the true causal effect of X on Y by comparing the so-called *post-interventional distribution* $P_{\mathcal{M}[X \leftarrow x]}(Y)$ at different values x, where $\mathcal{M}[X \leftarrow x]$ is the SCM obtained from $\mathcal{M}$ by applying $X \leftarrow x$.[2] By 'disconnecting" X from any of its possible causes, interventions prevent any source of spurious association between X and Y (Glymour et al., 2016) (i.e., caused by variables other than X and that are not descendants of X).[3] In the following we will use the notation I (and $\mathcal{M}[I]$) to denote a set of interventions $I = \{V_i \leftarrow v_i\}_i$.

Counterfactuals. Upon observing a particular realization $\mathbf{v}$ of the SCM variables $\mathbf{V}$, counterfactuals answer the following question: *what would have been the value of some variable Y for observation $\mathbf{v}$ if we had applied intervention I on our model $\mathcal{M}$*? This corresponds to evaluating $\mathbf{V}$ in a hypothetical world characterized by the same context (i.e., same realization of random factors) that generated the observation $\mathbf{v}$ but under a different causal process.

Computing counterfactuals involves three steps (Glymour et al., 2016):

1. *abduction*: estimate the context given the observation, that is derive $P(\mathbf{U} \mid \mathbf{V} = \mathbf{v})$;
2. *action*: modify the SCM by applying the intervention of interest, for example $\mathcal{M}[I]$; and
3. *prediction*: evaluate $\mathbf{V}$ under the manipulated model $\mathcal{M}[I]$ and the inferred context.

We denote by $\mathcal{M}(\mathbf{v})[I]$ the *counterfactual model* obtained by replacing $P(\mathbf{U})$ with $P(\mathbf{U} \mid \mathbf{V} = \mathbf{v})$ in the SCM $\mathcal{M}$ and then applying intervention I. Note that here $\mathbf{v}$ is a realization of $\mathbf{V}$ under $\mathcal{M}$ and not under $\mathcal{M}[I]$.

As explained above, each observation $\mathbf{V} = \mathbf{v}$ can be seen as a deterministic function of a particular value u of $\mathbf{U}$. Therefore, the counterfactual model is deterministic too, assuming that such u can be identified from $\mathbf{V} = \mathbf{v}$. However, inferring u precisely is often not possible (as discussed later), resulting in a (non-Dirac) posterior distribution of contexts $P(\mathbf{U} \mid \mathbf{V} = \mathbf{v})$ and thus, a stochastic counterfactual value.

2.1.1. Causal effects

Estimating a causal effect amounts to comparing some variable Y (outcome, output) under different values of some other variable X (treatment, input). Interventions and counterfactuals enable this task by ruling out spurious association between X and Y, as discussed above. There are three main estimators of causal effects:

Individual Treatment Effect (ITE). For a context u, the ITE of $Y \in \mathbf{V}$ between interventions I_1 and I_0 is defined as $Y_{I_1}(u) - Y_{I_0}(u)$, where $Y_{I_i}(u)$ is the counterfactual value of Y induced by u under the post-intervention model $\mathcal{M}[I_i]$. As explained above, we don't have direct access to the exogenous values u but only to realizations $\mathbf{v} \sim \mathbf{P}_{\mathcal{M}}(\mathbf{V})$. Thus, below we define the ITE as a function of $\mathbf{v}$ (rather than u) by plugging in the average counterfactual value of Y w.r.t. the posterior $P(\mathbf{U} \mid \mathbf{V} = \mathbf{v})$:

$$ITE(Y, I_1, I_0, \mathbf{v}) = \mathbb{E}_{\mathcal{M}(\mathbf{v})[I_1]}[Y] - \mathbb{E}_{\mathcal{M}(\mathbf{v})[I_0]}[Y]. \quad (1)$$

Average Treatment Effect (ATE). ATE is used to estimate causal effects at the population level and is defined as the expected value (w.r.t. $P(\mathbf{U})$) of the individual treatment effect, or equivalently, as the difference of post-interventional expectations:

$$ATE(Y, I_1, I_0) = \mathbb{E}_{\mathcal{M}[I_1]}[Y] - \mathbb{E}_{\mathcal{M}[I_0]}[Y]. \quad (2)$$

Conditional Average Treatment Effect (CATE). The CATE is the conditional version of ATE. This estimator is useful when the treatment effect may vary across the population depending on the value of some variables V:

$$CATE(Y, I_1, I_0, v) = \mathbb{E}_{\mathcal{M}[I_1]}[Y \mid V = v] - \mathbb{E}_{\mathcal{M}[I_0]}[Y \mid V = v]. \quad (3)$$

2.2. Markov Decision Processes (MDPs)

MDPs are a class of stochastic models to describe sequential decision making processes, where at each step t, an agent in state s_i performs some action a_i determined by a policy π ending up in state $s_{i+1} \sim P(\cdot \mid s_i, a_i)$. The agent receives some reward $\mathcal{R}(s_i, a_i)$ for performing a_i at s_i. Here, we focus on MDPs with finite state and action spaces. Without loss of generality, we restrict the policy class to deterministic policies (Puterman, 2014). Moreover, each MDP state satisfies a (possibly empty) set of atomic propositions, with AP being the set of atomic propositions.

Definition 2. (Markov Decision Process (MDP)). *An MDP is a tuple $\mathcal{P} = (\mathcal{S}, \mathcal{A}, P_{\mathcal{P}}, P_I, \mathcal{R}, L)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the set of actions, $P_{\mathcal{P}} : (\mathcal{S} \times \mathcal{A} \times \mathcal{S}) \to [0,1]$ is the transition probability function, $P_I : \mathcal{S} \to [0,1]$ is the initial state distribution, $\mathcal{R} : (\mathcal{S} \times \mathcal{A}) \to \mathbb{R}$ is the reward function, and $L : \mathcal{S} \to 2^{AP}$ is a labelling function, which assigns to each state $s \in \mathcal{S}$ the set of atomic propositions that are valid in s. A (deterministic) policy π for $\mathcal{P}$ is a function $\pi : \mathcal{S} \to \mathcal{A}$.*

An agent acting under policy π in an MDP environment will induce an MDP path τ, as follows:

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

Definition 3. (MDP path). *A path* $\tau = (s_1, a_1), (s_2, a_2), \ldots$ *of an MDP* $\mathcal{P} = (\mathcal{S}, \mathcal{A}, P_{\mathcal{P}}, P_I, \mathcal{R}, L)$ *induced by a policy* π *is a sequence of state-action pairs where* $s_i \in \mathcal{S}$ *and* $a_i = \pi(s_i)$ *for all* $i \geq 1$. *The probability of a path* τ *is given by* $P_{\mathcal{P}}(\tau) = P_I(s_1) \cdot \prod_{i \geq 1} P_{\mathcal{P}}(s_{i+1} \mid s_i, a_i)$. *For a finite path* $\tau = (s_1, a_1), \ldots, (s_k, a_k)$, *we denote by* $Paths_{\mathcal{P},\pi}(\tau)$ *the set of all (infinite) paths with prefix* τ *induced by MDP* $\mathcal{P}$ *and policy* π, *which has probability* $P_{\mathcal{P}}(Paths_{\mathcal{P},\pi}(\tau)) = P_{\mathcal{P}}(\tau)$.

We denote by $|\tau|$ the length of the path, by $\tau[i]$ the i-th element of τ (for $0 < i \leq |\tau|$), by $\tau[i:]$ the suffix of τ starting at position i (inclusive), and by $\tau[i:i+j]$ the subsequence spanning positions i to $i+j$ (inclusive). Even though $\tau[i]$ denotes the pair (s_i, a_i) of the path, we will often use it, when the context is clear, to denote only the state s_i. We slightly abuse notation and write $Paths_{\mathcal{P},\pi}(s)$ to denote the set of paths induced by π and starting with s.

Usually, an MDP is stationary, meaning that its transition probability function and/or reward function remain fixed over time. However, there exists a variant, called a non-stationary MDP (Lecarpentier and Rachelson, 2019), where the transition probability function and/or reward function may change over time. A non-stationary MDP can be converted to a stationary MDP by augmenting its state space with a variable that keeps track of the time.

An MDP under a fixed policy can be described as a deterministic-time Markov Chain (DTMC), as follows.

Definition 4. (Induced DTMC). *An MDP* $\mathcal{P} = (\mathcal{S}, \mathcal{A}, P_{\mathcal{P}} P_I, R, L)$ *and a policy* $\pi : \mathcal{S} \to \mathcal{A}$ *induce a discrete-time Markov Chain (DTMC)* $\mathcal{D}^{\mathcal{P},\pi} = (\mathcal{S}, P_{\mathcal{D}^{\mathcal{P},\pi}}, P_I, R^{\mathcal{P},\pi}, L)$ *where for* $s, s' \in \mathcal{S}$, $P_{\mathcal{D}^{\mathcal{P},\pi}}(s' \mid s) = P_{\mathcal{P}}(s' \mid s, \pi(s))$, *and for* $s \in \mathcal{S}$, $R^{\mathcal{P},\pi}(s) = R(s, \pi(s))$. *Paths of* $\mathcal{D}^{\mathcal{P},\pi}$ *are sequences of states, and their probabilities are defined similarly to Def. 3.*

2.3. SCM-based encoding of MDPs

We now present the SCM-based encoding of MDPs introduced in (Oberst and Sontag, 2019). For a given path length T, the SCM $\mathcal{M}^{\mathcal{P},\pi,T}$ induced by an MDP $\mathcal{P}$ and a policy π characterizes the unrolling of paths of $\mathcal{P}$ of length T, that is it has endogenous variables S_t and A_t describing the MDP's state and action at each time step t, where $t = 1, \ldots, T$. These are defined by the structural equations:

$$S_{t+1} = f(S_t, A_t, U_t); \quad A_t = \pi(S_t); \quad S_1 = f_0(U_0), \tag{4}$$

where the probabilistic state transition at t, $P_{\mathcal{P}}(S_{t+1} \mid S_t, A_t)$, is encoded as a deterministic function f of S_t, A_t, and the (random) exogenous variables U_t, while the random choice of the initial state, $P_I(S_1)$, as a deterministic function f_0 of U_0.

We stress that the SCM encoding does not require any assumptions about the structure of the MDP: such encoding results in an acyclic graph, while the original MDP need not be. Figure 2 shows the causal diagram resulting from this SCM encoding.

Note that both $P_{\mathcal{P}}(S_{t+1} \mid S_t, A_t)$ and $P_I(S_1)$ are categorical distributions and encoding them in the above SCM form (i.e., as functions of a random variable) is not obvious. Oberst and Sontag (2019) proposed a solution termed *Gumbel-Max SCM*, as given by:

$$\begin{aligned} S_{t+1} &= f(S_t, A_t, U_t = (G_{s,t})_{s \in \mathcal{S}}) \\ &= \underset{s \in S}{arg\,max} \left\{ \log(P_{\mathcal{P}}(S_{t+1} = s \mid S_t, A_t)) + G_{s,t} \right\} \end{aligned} \tag{5}$$

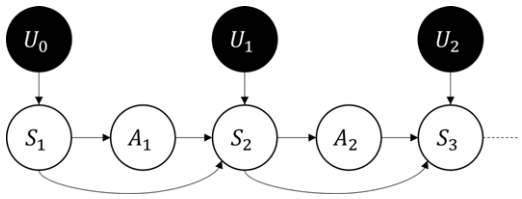


Figure 2. Causal diagram for the SCM encoding of an MDP. Black circles represents exogenous variables, while white circles represent endogenous ones.

where, for $s \in \mathcal{S}$ and $t \in 1 = \ldots, T$, $G_{s,t} \sim$ Gumbel. This is based on the Gumbel-Max trick, by which one can sample from a categorical distribution with $\mathcal{K}$ categories (corresponding to the $|S|$ MDP states in our case) by first drawing realizations $g_1, \ldots, g_{\mathcal{K}}$ of a standard Gumbel distribution and then by setting the outcome to $arg\,max_j \{\log(P(Y = j)) + g_j\}$. By using the Gumbel-Max trick, the assignment $S_{t+1} = f(S_t, A_t, (G_{s,t})_{s \in \mathcal{S}})$ in (5) will be equivalent to sampling $S_{t+1} \sim P_{\mathcal{P}}(S \mid S_t, A_t)$:

Proposition 1. (Gumbel-Max SCM correctness). *Given an MDP* $\mathcal{P}$, *policy* π, *and time bound* T, *then for any path* τ *of* $\mathcal{P}$ *induced by* π *of length* T, *we have* $P_{\mathcal{M}^{\mathcal{P},\pi,T}}(\tau) = P_{\mathcal{P}}(\tau)$, *where* $\mathcal{M}^{\mathcal{P},\pi,T}$ *is the Gumbel-Max SCM for* $\mathcal{P}$, π, *and* T.

Importantly, the *Gumbel-Max SCM* encoding enjoys a desirable property called *counterfactual stability*:

Definition 5. (Counterfactual stability (Oberst and Sontag, 2019)). *An SCM* $\mathcal{M}$ *satisfies counterfactual stability relative to a categorical variable* Y *of* $\mathcal{M}$ *if whenever we observe* $Y = i$ *under some intervention* I, *then the counterfactual value of* Y *under* $I' \neq I$ *remains* $Y = i$ *unless* I' *increases the relative likelihood of an alternative outcome* $j \neq i$, that is *unless* $P_{\mathcal{M}[I']}(Y{=}j)/P_{\mathcal{M}[I]}(Y{=}j) > P_{\mathcal{M}[I']}(Y{=}i)/P_{\mathcal{M}[I]}(Y{=}i)$.

Intuitively, the above definition tells us that, in a counterfactual scenario, we would observe the same outcome $Y = i$ unless the intervention increases the relative likelihood of an alternative outcome $Y = j$, that is, unless $\frac{p'_j}{p_j} > \frac{p'_i}{p_i}$ holds for some j.

Gumbel-Max SCMs are the most prominent encoding that can express categorical variables as functions of independent random variables and that satisfy counterfactual stability.[4] However, there also exists methods that generalise to other causal mechanisms with the counterfactual stability property (see Section 7).

Counterfactual inference. Given we observed an MDP path $\tau = (s_1, a_1), \ldots, (s_{|\tau|}, a_{|\tau|})$, counterfactual inference in this setting entails deriving $P((G_{s,t})_{s \in \mathcal{S}}^{t=1,\ldots,|\tau|-1} \mid \tau)$. Essentially, this means finding values for the Gumbel exogenous variables compatible with τ. By the Markov property, the above can be factorized as follows:

$$P((G_{s,t})_{s \in \mathcal{S}}^{t=1,\ldots,|\tau|-1} \mid \tau) = P((G_{s,1})_{s \in \mathcal{S}} \mid s_1) \cdot \prod_{t=2}^{|\tau|-1} P((G_{s,t})_{s \in \mathcal{S}} \mid s_t, a_t, s_{t+1}).$$

However, the mechanism of (5) is non-invertible, i.e., given s_t and a_t, there might be multiple values of $(G_{s,t})_{s \in \mathcal{S}}$ leading to the same s_{t+1}. This implies that *MDP counterfactuals can't be uniquely identified*, a problem that affects categorical counterfactuals in general and not just Gumbel-Max SCMs (Oberst and Sontag, 2019).

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

As suggested by Oberst and Sontag (2019), we can perform (approximate) posterior inference of $P((G_{s,t})_{s\in\mathcal{S}}|s_t,a_t,s_{t+1})$ through *rejection sampling*. This involves sampling from the prior $P((G_{s,t})_{s\in\mathcal{S}})$, and rejecting all the realizations $(g_{s,t})_{s\in\mathcal{S}}$ for which $f(s_t,a_t,(g_{s,t})_{s\in\mathcal{S}}) \neq s_{t+1}$.

Interventions in MDPs. In principle, we can consider any kind of intervention I over the SCM encoding of an MDP. Arguably, the most relevant case is when I affects the MDP policy π. For instance, in some applications, we might want to replace π with a more conservative or aggressive policy. Hence, in the following, we assume interventions of the form $I = \{(\pi \leftarrow \pi')\}$ for some policy π' (i.e., we change the RHS of the equation for A_t in the SCM (4)).

Example 1. (MDP counterfactuals). *Consider an MDP model of a light switch. The MDP has two states,* $\mathcal{S} = \{\mathsf{On}, \mathsf{Off}\}$, *and we can take two actions,* $\mathcal{A} = \{\mathsf{Switch}, \mathsf{Nop}\}$. *If we take action* Switch, *the state of the MDP changes (from* On, *to* Off, *or vice versa) with probability* 0.9, *and it remains the same with probability* 0.1. *If we take action* Nop, *with probability* 0.9 *the MDP's state does not change, and with probability* 0.1 *the state changes. We fix the following policy:* $\pi(\mathsf{On}) = \mathsf{NOP}$ *and* $\pi(\mathsf{Off}) = \mathsf{Switch}$.

Assume we observe the path $\tau = \mathsf{Off} \xrightarrow{\mathsf{Switch}} \mathsf{On} \xrightarrow{\mathsf{NOP}} \mathsf{Off}$, *where the first step has probability* 0.9 *and the second step* 0.1. *First, we want to show that the Gumbel-max SCM formulation (5) yields the same probability values, modulo sampling variability. In Figure 3a and Figure 3b we show the values of* $\log(P_{\mathcal{P}}(\mathsf{Off}\,|S_t, A_t)) + G_{\mathsf{Off},t}$ *(x-axis) and* $\log(P_{\mathcal{P}}(\mathsf{On}\,|S_t, A_t)) + G_{\mathsf{On},t}$ *(y-axis) obtained by sampling* 1000 *realizations of the Gumbel variables* $\mathbf{G}$. *We see indeed that, at* $t = 2$, *89.7% of these points lie above the identity line, that is they yield* On *as the next state. At* $t = 3$, *we find that 10.9% of the points yield* Off *as the next state.*

In Figure 3c and Figure 3d, we show the computation of counterfactuals. Assume an intervention that changes the policy into one that constantly performs action Switch. *Now, we want to see what is the probability of path* $\tau' = \mathsf{Off} \xrightarrow{\mathsf{Switch}} \mathsf{On} \xrightarrow{\mathsf{Switch}} \mathsf{Off}$ *given that we observed* τ. *That is, we compute the probability of* τ' *in the counterfactual SCM model where the (prior) Gumbel variables are replaced by* $\mathbf{G}' = \mathbf{G} \mid \tau$, that is *those inferred from* τ.

First note that τ *and* τ' *perform the same first step. Hence, this step has probability* 1 *under* $\mathbf{G}'$ *because* $\mathbf{G}'$ *is defined such that it assigns probability* 1 *to the observed path (see also Proposition 3 for a similar statement). In the second step, the observed path* τ *transitioned into* Off *after performing* Nop, *despite a probability of* 0.9 *of jumping into* On. *This means that* $\mathbf{G}'$ *strongly favours* Off *(over* On*) to happen in the second step. Hence, we expect that the probability of* On $\xrightarrow{\mathsf{Switch}}$ Off *in the counterfactual world will be higher than the nominal probability* $P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{Switch})$. *In particular, by counterfactual stability (Def. 5), such probability should be* 1 *because the intervention doesn't make state* On *more likely to happen (rather the opposite: the relative likelihood of* On *is indeed* 0.1/0.9, *while it is* 0.9/0.1 *for* Off*). This can be proven also by showing that, by rejection sampling, we have that:*

$$P_{\mathbf{G}'}\Big(\log(P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{Nop})) + G'_{\mathsf{Off},t} > \log(P_{\mathcal{P}}(\mathsf{On} \mid \mathsf{On}, \mathsf{Nop})) + G'_{\mathsf{On},t}\Big) = 1.$$

Since $0.9 = P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{Switch}) > P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{NOP}) = 0.1$ *and* $0.1 = P_{\mathcal{P}}(\mathsf{On} \mid \mathsf{On}, \mathsf{Switch}) < P_{\mathcal{P}}(\mathsf{On} \mid \mathsf{On}, \mathsf{NOP}) = 0.9$, *it follows that*

$$P_{\mathbf{G}'}\Big(\log(P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{Switch})) + G'_{\mathsf{Off},t} > \log(P_{\mathcal{P}}(\mathsf{On} \mid \mathsf{On}, \mathsf{Switch})) + G'_{\mathsf{On},t}\Big) = 1,$$

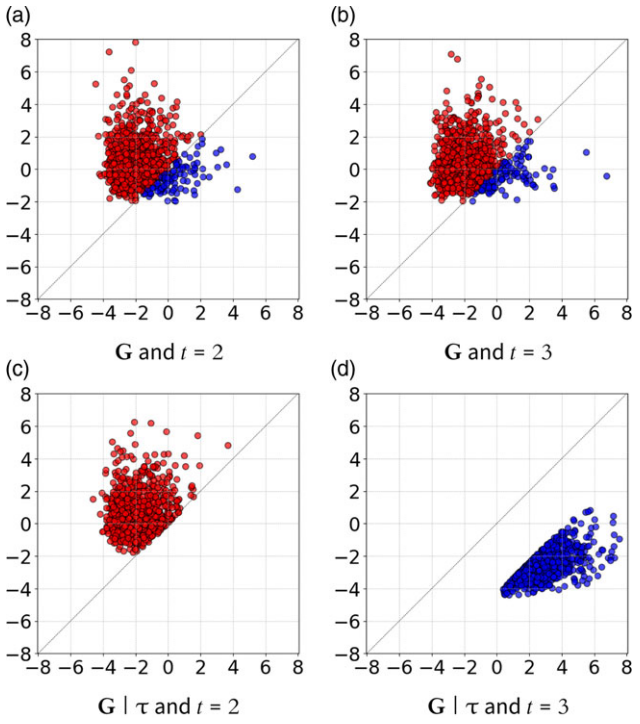


Figure 3. Light switch MDP (example 1). X-axis: $\log(P_{\mathcal{P}}(\mathsf{Off}|S_t, A_t)) + G_{\mathsf{Off},t}$; Y-axis: $\log(P_{\mathcal{P}}(\mathsf{On}|S_t, A_t)) + G_{\mathsf{On},t}$. Plots (a) and (b) are relative to the prior Gumbel $\mathbf{G}$ and the observed path τ (using 1000 realizations for $\mathbf{G}$). Plots (c) and (d) are relative to the posterior Gumbel $\mathbf{G}\,\tau$ and the counterfactual path τ'. Points leading to state On are in red, while those for Off are in blue.

i.e., performing action Switch *at state* On *has probability* 1 *of leading into state* Off *in the counterfactual world. In particular, since* $P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{Switch}) > P_{\mathcal{P}}(\mathsf{Off} \mid \mathsf{On}, \mathsf{NOP})$, *the points in Figure 3d (corresponding to the counterfactual step) are shifted to the right compared to Figure 3b (observed step).*

3. Construction of counterfactual MDP

Consider a Gumbel-max SCM $\mathcal{M}^{\mathcal{P}}$ for an MDP $\mathcal{P}$ under policy π, and a (finite) path τ of $\mathcal{M}^{\mathcal{P}}$. Let $\mathbf{G}' = (G'_{s,i})_{s\in\mathcal{S}}^{i=1,\ldots,|\tau|-1}$ be the set of *posterior Gumbel variables*, where, for $i = 1, \ldots, |\tau| - 1$, $G'_{s,i} \sim P_{\mathcal{M}^{\mathcal{P}}}(G_{s,i}|\tau)$ and $G_{s,i} \sim$ Gumbel. That is, $G'_{s,i}$ is the value of the exogenous variable (associated to position i and state s) inferred from τ. Then, for $i = 1, \ldots, |\tau| - 1$, we have the following transition probability function, which directly follows from the SCM (5):

$$P_{\mathcal{P},i,\tau}(s' \mid s, a) = \Pr_{(G'_{s'',i})_{s''\in\mathcal{S}}}\left(s' = \arg\max_{s''\in S}\left\{\log(P_{\mathcal{P}}s'' \mid s, a)) + G'_{s'',i}\right\}\right). \quad (6)$$

See also (Tsirtsis, De, and Rodriguez, 2021) for a similar definition. Then, we can express this non-stationary MDP as a stationary one by augmenting its state space as follows.

Definition 6. (Counterfactual MDP). *Given an MDP* $\mathcal{P}$, *policy* π, *and a finite path* τ *of* $\mathcal{P}$ *under* π, *the corresponding (stationary) counterfactual MDP* $\mathcal{P}^{\tau} = (\mathcal{S}^{\tau}, \mathcal{A}, P_{\mathcal{P}}\tau, P_I^{\tau}, \mathcal{R}', L')$. *Here,* $\mathcal{S}^{\tau} = \mathcal{S} \times \{1, \ldots, |\tau|\}$ *is an augmented state space where each state* $s' \in \mathcal{S}^{\tau}$ *corresponds to a tuple* $s' = (s, i)$, *where each state* $s \in \mathcal{S}$ *from the nominal MDP* $\mathcal{P}$ *has been augmented with a timestep* i, $\mathcal{R}' : (\mathcal{S}^{\tau} \times \mathcal{A}) \to \mathbb{R}$ *is a reward function such that* $\mathcal{R}'((s, i), a) = \mathcal{R}(s,\ a)$, $L' : \mathcal{S}^{\tau} \to 2^{AP}$

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

is a labelling function such that $L'((s,i)) = L(s)$, $P_I^\tau(\tau[1],1) = 1$, *and for any* $(s,i),(s',i') \in \mathcal{S}^\tau$ *and* $a \in \mathcal{A}$,

$$P^\tau_{\mathcal{P}}(s',i' \mid s,i,a) = \begin{cases} P_{\mathcal{P}}(s' \mid s,a) & \text{if } i = i' = |\tau| \\ P_{\mathcal{P},i,\tau}(s' \mid s,a) & \text{if } i < |\tau| \text{ and } i' = i+1 \\ 0 & \text{otherwise} \end{cases}.$$

In other words, in $\mathcal{P}^\tau$ we introduce an extra variable to track the position i of the observed path τ. Then, for $i < |\tau|$, $\mathcal{P}^\tau$ behaves according to the transition probabilities of the counterfactual model, as per Eq. 6. For $i = |\tau|$, $\mathcal{P}^\tau$ is equivalent to the original MDP model $\mathcal{P}$, because we do not have an observation on which we can condition our Gumbel exogenous variables. Also, P_I^τ is defined such that $\mathcal{P}^\tau$ admits only one initial state, that is, the first state of τ. The following proposition shows that the counterfactual MDP reduces to the original MDP in the special case when $|\tau| = 1$.

Proposition 2. *If* $|\tau| = 1$, *then the counterfactual MDP* $\mathcal{P}^\tau$ *of an MDP* $\mathcal{P}$ *is equivalent to* $\mathcal{P}(\tau[1])$.

Proof. It is easy to see that, by applying Def. 6, we recover the definition of the original MDP $\mathcal{P}$ (with the provision that $\mathcal{S}^\tau = \mathcal{S} \times \{1\}$) initialised at $\tau[1]$, the only state of τ. Indeed, if τ contains only one state, then we do not have any observed transitions to perform posterior inference of the Gumbel exogenous variables. □

Another useful property is that if we do not perform any interventions, that is we maintain the original policy π, then the counterfactual MDP induces the observed path τ with probability 1, as expected.

Proposition 3. *Given* $\mathcal{P}$, π, *and* τ *as per Definition 6, then the resulting counterfactual MDP* $\mathcal{P}^\tau$ *is such that* $P_{\mathcal{P}^\tau}(\tau) = 1$.

Proof. It is enough to show that, for any $1 \le i < |\tau|$, it holds that

$$P_{\mathcal{P},i,\tau}(s_{i+1} \mid s_i, a_i) = \Pr_{(G'_{s'',i})_{s'' \in \mathcal{S}}} \left(s_{i+1} = \arg\max_{s'' \in S} \left\{ \log(P_{\mathcal{P}}(s'' \mid s_i, a_i)) + G'_{s'',i} \right\} \right) = 1.$$

This is true because the posterior Gumbel variables $G'_{s'',i}$ are inferred in order to be consistent with the observed path. This holds also for (approximate) inference via rejection sampling: since we discard all the Gumbel realizations incompatible with the observation, we have that

$$\Pr_{(G'_{s'',i})_{s'' \in \mathcal{S}}} \left(s_{i+1} \neq \arg\max_{s'' \in S} \left\{ \log(P_{\mathcal{P}}(s'' \mid s_i, a_i)) + G'_{s'',i} \right\} \right) = 0,$$

which proves the above equality. □

In the following, for simplicity, we will use policies π defined over $\mathcal{S}$ (the state space of the original MDP $\mathcal{P}$) also for the augmented state space $\mathcal{S}^\tau$ of the counterfactual MDP, by assuming $\pi(s,i) = \pi(s)$ for any i.

4. PCFTL: a probabilistic temporal logic with interventions, Counterfactuals, and Causal Effects

In this section, we formally define *PCFTL (Probabilistic CounterFactual Temporal Logic)*. A PCFTL formula is interpreted over an MDP $\mathcal{P}$, a policy π, and an observed path τ resulting from $\mathcal{P}$ and π.

PCFTL extends PCTL* (Baier et al., 1997; Baier, 1998) with a *counterfactual operator* $I_{@t}.P_{\bowtie p}(\phi)$, a *counterfactual reward operator* $I_{@t}.R^{\lesseqgtr \hbar}_{\bowtie r}$, and two *causal effect operators*, $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p}(\phi)$ and $\Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr \hbar}_{\bowtie r}$. The latter two formulas are defined as the difference of counterfactual probabilities (resp., cumulative rewards) between interventions I_1 and I_0, in line with the definition of treatment effects in Section 2.1.

PCFTL syntax. The syntax of PCFTL is as follows:

$$\Phi ::= \top \mid \rho \mid \neg\Phi \mid \Phi \wedge \Phi \mid I_{@t}.P_{\bowtie p}(\phi) \mid I_{@t}.R^{\lesseqgtr k}_{\bowtie r} \mid \Delta^{I_1,I_0}_{@t}.P_{\bowtie p'}(\phi) \mid \Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr k}_{\bowtie r}$$
$$\phi ::= \Phi \mid \neg\phi \mid \phi \wedge \phi \mid \phi \mathcal{U}_{[a,b]} \phi$$

where I, I_0, I_1 are (possibly empty) interventions, $t \in \mathbb{Z}^{\geq 0}$, $\rho \in AP$, $p \in [0,1]$, $r \in \mathbb{R}$, $p' \in [-1,1]$, $\bowtie \in \{<, \leq, \geq, >\}$, $\hbar \in \mathbb{Z}^{\geq 1}$, and $[a,b]$ is an interval with $a \in \mathbb{Z}^{\geq 0}$ and $b \in \mathbb{Z}^{\geq 0} \cup \{\infty\}$. State formulas Φ can be atomic propositions, counterfactual or causal effect formulas, or logical combinations of them. Path formula $\phi_1 \mathcal{U}_{[a,b]} \phi_2$ is satisfied by paths where ϕ_2 holds at some time point within the (potentially unbounded) interval $[a,b]$ and ϕ_1 always holds before that point. Other standard bounded temporal operators are derived as: $\mathcal{F}_{[a,b]}\phi \equiv \top \mathcal{U}_{[a,b]}\phi$ (*eventually*), $\mathcal{G}_{[a,b]}\phi \equiv \neg\mathcal{F}_{[a,b]}\neg\phi$ (*always*), and $\mathcal{X}\phi \equiv \mathcal{F}_{[1,1]}\phi$ (*next*).

Before introducing the semantics of PCFTL, we define the quantitative counterfactual operators $I_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau)$ and $I_{@t}.R^{\lesseqgtr \hbar}_{=?}(\mathcal{P},\pi,\tau)$. These quantify the probability of a path formula ϕ (resp., the expected cumulative reward up to step $\hbar$) in the counterfactual model obtained from MDP $\mathcal{P}$, given that we observed path τ under policy π, and by applying I from t steps back in the past (we emphasise that t is a local indexing).

$$I_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau) = P_{\mathcal{P}'}(\{\tau' \in Paths_{\mathcal{P}',\pi'} \mid (\mathcal{P}',\pi',\tau',1) \models \phi\}) \tag{7}$$

$$I_{@t}.R^{\lesseqgtr k}_{=?}(\mathcal{P},\pi,\tau) = \sum_{\tau' \in Paths_{\mathcal{P}',\pi'}} \left(\mathcal{P}_{\mathcal{P}'}(\tau') \cdot \sum_{i=1}^{\hbar} \mathcal{R}(\tau'[i]) \right) \tag{8}$$

where $\mathcal{P}' = \mathcal{P}^{\tau[|\tau|-t:]}$ is the counterfactual MDP derived from $\mathcal{P}$ and $\tau[|\tau|-t:]$, i.e., the path suffix starting at the time of intervention, and π' is the intervention policy (corresponding to π if $I = \emptyset$). Note that the probability of ϕ is evaluated in the counterfactual model starting from the time of intervention, not from the last state of the path (to do so, one can simply replace ϕ with $\mathcal{F}_{[t,t]}\phi$). The satisfaction relation for path formulae is as follows.

Definition 7. (Semantics of PCFTL). *Given a PCFTL formula* Φ, *an MDP* $\mathcal{P}$, *and a path* τ *of* $\mathcal{P}$ *under some policy* π, *the PCFTL satisfaction relation* $\models$ *is defined by the following rules:*

$$\begin{array}{lcll}
(\mathcal{P},\pi,\tau) & \models & \rho & \text{if } \rho \in L(\tau[|\tau|]) \\
(\mathcal{P},\pi,\tau) & \models & \neg\Phi & \text{if } (\mathcal{P},\pi,\tau) \not\models \Phi \\
(\mathcal{P},\pi,\tau) & \models & \Phi_1 \wedge \Phi_2 & \text{if } ((\mathcal{P},\pi,\tau) \models \Phi_1) \wedge ((\mathcal{P},\pi,\tau) \models \Phi_2) \\
(\mathcal{P},\pi,\tau) & \models & I_{@t}.P_{\bowtie p}(\phi) & \text{if } I_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau) \bowtie p \\
(\mathcal{P},\pi,\tau) & \models & I_{@t}.R^{\lesseqgtr k}_{\bowtie r} & \text{if } I_{@t}.R^{\lesseqgtr k}_{=?}(\mathcal{P},\pi,\tau) \bowtie r \\
(\mathcal{P},\pi,\tau) & \models & \Delta^{I_1,I_0}_{@t}.P_{\bowtie p'}(\phi) & \text{if } (I_{1@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau) - I_{0@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau)) \bowtie p' \\
(\mathcal{P},\pi,\tau) & \models & \Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr k}_{\bowtie r} & \text{if } (I_{1@t}.R^{\lesseqgtr k}_{=?}(\mathcal{P},\pi,\tau) - I_{0@t}.R^{\lesseqgtr k}_{=?}(\mathcal{P},\pi,\tau)) \bowtie r \\
 & & & \\
(\mathcal{P},\pi,\tau,t) & \models & \Phi & \text{if } (\mathcal{P},\pi,\tau[1:t]) \models \Phi \\
(\mathcal{P},\pi,\tau,t) & \models & \neg\phi & \text{if } (\mathcal{P},\pi,\tau,t) \not\models \phi \\
(\mathcal{P},\pi,\tau,t) & \models & \phi_1 \wedge \phi_2 & \text{if } ((\mathcal{P},\pi,\tau,t) \models \phi_1) \wedge ((\mathcal{P},\pi,\tau,t) \models \phi_2) \\
(\mathcal{P},\pi,\tau,t) & \models & \phi_1 \mathcal{U}_{[a,b]} \phi_2 & \text{if } \exists t_1 \in [a,b].((\mathcal{P},\pi,\tau,t+t_1) \models \phi_2 \wedge \\
 & & & \quad \forall t_2 \in [0,t_1).((\mathcal{P},\pi,\tau,t+t_2) \models \phi_1)).
\end{array}$$

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

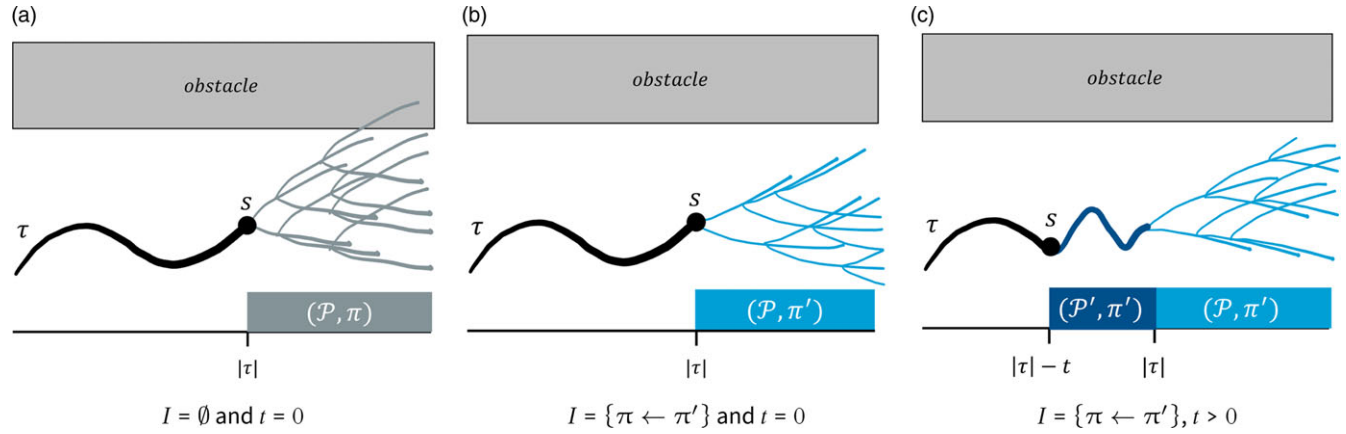


Figure 4. Three scenarios for the evaluation of $I_{@t}.P_{\bowtie p}(\phi)$. The observed path τ is in black. The counterfactual path (induced by the counterfactual MDP $\mathcal{P}' = \mathcal{P}^{\tau[|\tau|-t:]}$ and the intervention policy π') is in dark blue (in general we have a distribution of such paths, but here we show only one for simplicity). Paths extensions under the nominal policy π are in gray, and those under π' in light blue. The horizontal axis represents time (or path positions), and the vertical axis the MDP state (continuous and one-dimensional for illustration purposes). While none of the three examples hit the obstacle within the observed/counterfactual path, moving forward, π yields a higher probability of this happening.

Remark 1. *A main difference compared to existing temporal logics like* PCTL* *is that a PCFTL formula Φ is evaluated over a path of observed states and actions rather than the current state only. Keeping track of the past allows us to perform counterfactual reasoning; see Equations 7 and 8. Without counterfactuals, there would be no need to carry over the path, but only the current state because the system is Markovian.*[5] *Also,* PCTL* *formulas evaluated over a DTMC model, while in our logic, it is convenient to keep $\mathcal{P}$ and π separated rather than working with the DTMC induced by $\mathcal{P}$ and π.*

Remark 2. *Normally, probabilistic model checking of MDPs is concerned with computing the maximum or minimum satisfaction probability across the policy space (Baier and Katoen, 2008). In this work, we instead want to compute probabilities w.r.t. given nominal and interventional/counterfactual policies, not across the entire policy space.*

Building on the intuition that our counterfactual operator generalizes PCTL*'s probabilistic operator, we demonstrate below that our logic subsumes PCTL*.

Proposition 4. *Every* PCTL* *formula is a PCFTL formula, but not viceversa.*

Proof. It suffices to prove that PCTL*'s probabilistic operator (see Baier and Katoen, 2008) is a special case of our counterfactual operator. Path formulas and their semantics are indeed equivalent between the two logics, with the only difference being that in PCFTL we keep track of the point t in the path at which ϕ is evaluated.

In particular, we show that, for $s \in \mathcal{S}$, $P_{=?}(\phi)(\mathcal{P}, \pi, s) = \emptyset_{@0}.P_{=?}(\phi)(\mathcal{P}, \pi, (s))$, where $P_{=?}(\phi)(\mathcal{P}, \pi, s) = P_{\mathcal{P}(s)}(\{\tau' \in Paths_{\mathcal{P}(s),\pi} | (\mathcal{P}(s), \pi, \tau', 1) \models \phi\})$ is the quantitative probabilistic operator. By applying (7), we have that

$$\emptyset_{@0}.P_{=?}(\phi)(\mathcal{P}, \pi, (s)) = P_{\mathcal{P}'}(\{\tau' \in Paths_{\mathcal{P}',\pi'} \mid (\mathcal{P}', \pi', \tau', 1) \models \phi\})$$

where $\pi' = \pi$ (the intervention is empty), and $\mathcal{P}' = \mathcal{P}^{(s)}$. By Proposition 2, we have that $\mathcal{P}^{(s)} = \mathcal{P}(s)$. □

Expressiveness. We discuss the counterfactual operator $I_{@t}.P_{\bowtie p}(\phi)$ (a similar reasoning holds for $I_{@t}.R^{\mathcal{F}^R_r}_{\bowtie r}$). When $t = 0$, our operator captures the post-interventional probability; that is, the probability of a path formula ϕ after we apply intervention I at the current state. In this case, no counterfactuals need to be inferred because, trivially, we don't have any observed MDP states beyond the time of intervention (see Figure 4b). Indeed, by Proposition 2, we have that $\mathcal{P}^{\tau[|\tau|-0:]} = \mathcal{P}(\tau[|\tau|-0]) = \mathcal{P}(\tau[|\tau|])$, that is the counterfactual MDP conditioned on the last state of τ corresponds to the original MDP $\mathcal{P}$ initialized at that state. For this reason, as also shown in the proof of Propositon 4, our operator subsumes PCTL*'s probabilistic formula (which is indeed omitted in PCFTL): when $t = 0$ and $I = \emptyset$, $I_{@t}.P_{\bowtie p}(\phi)$ corresponds to evaluating $P_{\bowtie p}(\phi)$ w.r.t. the original MDP $\mathcal{P}$ initialized at $\tau[|\tau|-0] = \tau[|\tau|]$ and under the original policy π (see Figure 4a). Thus, $\emptyset_{@0}.P_{\bowtie p}(\phi) \equiv P_{\bowtie p}(\phi)$.

When $t > 0$, our operator expresses a counterfactual query, which answers the question: given that we observed τ, what would have been the probability of ϕ if we had applied a particular intervention I at t steps back in the past (but under the same random circumstances that led to τ)? A common choice is to apply I at the beginning of τ ($t = |\tau| - 1$) but other options are possible, for example intervening before some violation has happened in τ. We stress, however, that our operator goes beyond the usual notion of counterfactuals, by which the outcomes of interest are obtained only from the observed (or counterfactual) path. Indeed, depending on the bounds in the temporal operators of ϕ, evaluating ϕ might require paths that extend beyond τ. Hence, up to the length of τ, ϕ is evaluated on counterfactual paths; beyond that point, paths follow the original MDP model $\mathcal{P}$ (which is precisely how our counterfactual MDP is constructed, see Def. 6) because there are no observations to condition on. We show why this matters in Example 2 below.

Example 2. *Consider an MDP $\mathcal{P}$ and an obstacle avoidance property $\varphi_H = \mathcal{G}_{[0,H]}\neg$*obstacle *for some horizon $H > 0$. Let τ be an observed path of $\mathcal{P}$ under some policy π. Let τ_I, with $|\tau_I| = |\tau|$, denote the counterfactual path obtained from τ by applying some intervention $I = \{\pi \leftarrow \pi'\}$ at the start. (For simplicity, we assume that only one counterfactual path is possible.) Now suppose that no obstacle is hit in τ or τ_I. So, in usual counterfactual analysis, one*

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

would conclude that the nominal policy and the intervention policy are equivalent relative to property φ_H and observation τ. However, if the safety property bound H extends beyond the length of τ, then it is necessary to reason about the future evolution of the MDP beyond τ (or τ_I): in one case, starting from the last state of τ and under the nominal policy; in the other, from the last state of the counterfactual path τ_I and under I's policy. At this point, it is entirely possible that going forward from the counterfactual world yields a higher probability of obstacle avoidance than remaining with the nominal policy, as illustrated in Figures 4a and 4c. Thus, limiting the analysis to outcomes within the observed/counterfactual past, as done in previous work (Oberst and Sontag, 2019; Tsirtsis, De, and Rodriguez, 2021), would lead to the wrong conclusion that the two policies are equivalent safety-wise.

Encoding treatment effects. We explain how the introduced causal effect operators $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p'}(\phi)$ and $\Delta^{I_1,I_0}_{@t}.R^{\not R}_{\bowtie r}$ can be used to express the traditional CATE and ITE estimators (defined in Section 2.1). We saw that CATE is the difference of post-interventional probabilities, conditioned on a particular value $V = v$ of some variable V. In reinforcement learning with MDPs, one sensible choice is to condition on the first state of the post-interventional path (Oberst and Sontag, 2019). Therefore, for the same argument made above about defining post-interventional probabilities with $I_{@0}.P_{\bowtie p}(\phi)$ formulas, we can express this notion of CATE in PCFTL with the formula $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p}(\phi)$. The latter indeed is the effect in the probability of ϕ between interventions I_1 and I_0, conditioned on paths starting with $\tau[|\tau|{-}0] = \tau[|\tau|]$ (the last state of τ).

ATE, the unconditional version of CATE, cannot be directly expressed in PCFTL because our semantics is defined over a non-empty path τ, and hence, probabilities are implicitly conditional on the last state $\tau[|\tau|]$. An equivalent of ATE can be defined as the expected value of the CATE formula $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p}(\phi)$ evaluated at the initial states $S \sim P_I(S)$ of the MDP.

Finally, akin to how $I_{@t}.P_{\bowtie p}(\phi)$ with $t > 0$ expresses a counterfactual probability (as discussed previously), the operator $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p}(\phi)$ with $t > 0$ provides a notion of ITE, because, like ITE, our operator is defined as the difference of the counterfactual probabilities $I_{1@t}.P_{=?}(\phi)$ and $I_{0@t}.P_{=?}(\phi)$.

4.1. Example properties

Below, we provide examples of useful properties that can be expressed with the newly introduced counterfactual and causal effect operators of PCFTL, for the verification of cyber-physical systems.

Example 3. *Let τ denote an observed path (of length τ) in an arbitrary MDP $\mathcal{P}$ under policy π. Let π' represent an alternative policy that we can intervene on, defined by $I' = \pi \leftarrow \pi'$, and let ϕ be a path formula describing some requirement of interest. Using PCFTL, we can express many interventional and counterfactual properties related to cyber-physical systems, such as:*

- Safety:
  - $I_{@|\tau|-1}.P_{\geq 0.99}(\mathcal{G}_{[0,20]}\text{signal} < \text{threshold})$: *'If we had replaced the nominal policy π with π' at the beginning, would the probability of the signal remaining below a specified safety threshold over the next 20 steps have been at least 99%?"*
  - $\Delta^{I',\emptyset}_{@0}.P_{>0}(\mathcal{G}_{[a,b]}\phi)$: *"Is π' safer than π moving forward from the current state (between bounds a and b)?" (this is a CATE-like query)*
  - $\emptyset_{@t}.P_{<p}(\mathcal{G}_{[a,b]}\phi) \rightarrow I'_{@t}.P_{\geq p}(\mathcal{G}_{[a,b]}\phi)$: *"Had we deployed π' t steps in the past, would we have observed a safety probability of at least p if π failed to achieve so?"*
  - $I'_{@t}.P_{=?}(\mathcal{F}_{[t',t']}(\neg\phi \wedge \Delta^{I'',\emptyset}_{@0}.P_{>0}\,\mathcal{F}_{[1,H]}\phi))$, *where* $I'' = \{(\pi \leftarrow \pi'')\}$ *and* $H \geq 1$: *"What would have been the probability, had we applied π' t steps in the past, of observing a violation after time t', and subsequently, of a different policy π'' yielding a better recovery probability than π'?"*
- Liveness:
  - $I_{@|\tau|-1}.P_{\geq 0.99}(\mathcal{G}_{[0,20]}\text{waiting_for_resource} < \mathcal{F}\,\text{acquired_resource})$: *'If we had replaced the nominal policy π with π' at the beginning, would the probability of avoiding resource starvation over the next 20 steps have been at least 99%?"*
- Reachability:
  - $I_{@0}.P_{\geq 0.95}(\mathcal{F}_{[0,10]}\text{goal})$: *"If we apply the intervention $I' = \{(\pi \leftarrow \pi')\}$ in the current state, will the probability of reaching the goal state(s) within 10 steps be at least 95%?"*
  - $\Delta^{I',\emptyset}_{@0}.P_{\geq 0}(\mathcal{F}_{[0,10]}\text{goal})$: *"If we replaced the nominal policy π with π' at the current time step, would this increase the likelihood of reaching the goal state(s) within the next 10 steps?"*
- Reward-based properties:
  - $I_{@10}.R^{\leq|\tau|}_{\geq 200}$: *"If we replaced the nominal policy π with π' in the last 10 time steps, would the expected reward be over 200?"*
  - $\Delta^{I',\emptyset}_{@|\tau|-1}.R^{\leq|\tau|}_{\geq 30}$: *"If we replaced the nominal policy π with π' at the beginning, would the expected reward over τ steps under π' have been at least 30 higher than the expected reward under π"*

4.2. Decidability

Despite the added expressiveness, PCFTL remains *decidable*. First, we note that the transition probability function of a counterfactual MDP, defined in (6), is a well-defined probability measure. Therefore, the set of paths induced by a counterfactual MDP $\mathcal{P}'$ and some policy is also measurable (Baier and Katoen, 2008), which ensures that we can quantify the probability of a path formula.

A decision procedure for PCFTL can be adapted from the standard model checking algorithm for a DTMC $\mathcal{D} = (S, P_{\mathcal{D}}, P_I, R, L)$ and a PCTL* formula Φ (Baier and Katoen, 2008), which we summarise next. The procedure traverses the parse tree of Φ bottom-up. For each node, representing a subformula Ψ, the satisfaction set $\text{Sat}(\Psi) = \{s \in S \mid s \vDash \Psi\}$ is computed. When Ψ is a simple Boolean formula, computing $\text{Sat}(\Psi)$ is straightforward, so we focus on the case $\Psi = P_{\bowtie p'}(\phi)$. Here, all maximal state subformulas of ϕ are replaced with new atomic propositions representing their satisfaction sets. This step is possible because the satisfaction sets are precomputed during the bottom-up traversal. This operation effectively transforms ϕ into an LTL property, which enables the computation of $P_{\mathcal{D}}(s \vDash \phi)$ using a standard automata-based approach (Baier and Katoen, 2008). Hence, we can compute the satisfaction set of Ψ as $\text{Sat}(\Psi) = \{s \in S \mid P_{\mathcal{D}}(s \vDash \phi) \bowtie p'\}$.

Model-checking PCTL* has double-exponential time complexity in $|\phi|$ due to the transformation of ϕ' into a deterministic Rabin automaton and polynomial complexity in the size of the DTMC. Moreover, as demonstrated by Kwiatkowska et al. (2007),

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

determining reward properties does not impact the decidability or time complexity of the model-checking procedure, so we will not discuss this case here.

The decision procedure for PCFTL follows a similar approach. We do not discuss Boolean and reward properties and cover the case when $\Psi = I_{@t}.P_{\bowtie p}(\phi)$ (from which a procedure for $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p'}(\phi)$ can be easily derived). The key difference here is that the satisfaction set for Sat(Ψ) cannot include states, but it must include paths because the satisfaction of $I_{@t}.P_{\bowtie p}(\phi)$ depends on an (observed) path. It is important to note that this set will include paths of at most length T where T is the largest t offset of an intervention appearing in any state subformula. Indeed, it is easy to see that the satisfaction of $I_{@t}.P_{\bowtie p}(\phi)$ w.r.t. a path τ (with $|\tau| \geq t$) depends only on the t-length suffix of τ (which is the suffix used to construct the counterfactual MDP, see (7)). To transform the path formula ϕ into an equivalent LTL formula (as done above), we now need to express these satisfaction sets (defined over finite paths, i.e., sequences of states) as atomic propositions (defined over states). This is possible by augmenting the MDP with memory to keep track of the last $T-1$ visited states.[6] In this way, there is a direct correspondence between the elements of Sat(Ψ) and the states of the augmented MDP, as desired. So, we can now construct our sets as done for the PCTL* case above, as $\text{Sat}(\Psi) = \{\tau \in \bigcup_{1 \leq i \leq T} \mathcal{S}^i \mid P_{\mathcal{D}^{\tau,\pi'}}(\tau[1] \models \phi)\}$ where $\mathcal{D}^{\tau,\pi'}$ is the (counterfactual) DTMC induced by the interventional policy π' and by the counterfactual MDP associated to the original MDP and path τ. Having shown that PCFTL model checking reduces to PCTL* model checking, its complexity is still polynomial in the size of the induced (counterfactual) DTMC, as the state space size of the augmented model is polynomial in that of the induced DTMC.

5. PCFTL verification with statistical model checking

We use statistical model checking (SMC) (Younes and Simmons, 2006; Legay et al., 2010) to determine whether our properties are satisfied, that is by sampling finite paths of the (counterfactual) MDP model. We leave the study of numerical-symbolic algorithms for future work.

Since we deal with finite paths, we consider a fragment of the logic with bounded temporal operators. Also, we restrict to non-nested properties, that is those where path sub-formulas ϕ do not contain in turn counterfactual operators (even though we allow for arbitrary nesting of temporal operators in ϕ). The complication with nested formulas is that we require multiple executions to determine the satisfaction of ϕ, leading to a sample size that is exponential in the depth of the nested operator (Younes and Simmons, 2006; Legay et al., 2010). Nevertheless, the fragment we consider is rich enough to express a variety of reinforcement learning tasks (see Section 6) and subsumes Probabilistic Bounded LTL (Zuliani, Platzer, and Clarke, 2013) (because our counterfactual formulas generalize probabilistic ones).

In short, with SMC we reduce the problem of checking $I_{@t}.P_{\geq p}(\phi)$ to one of hypothesis testing, given a sample of MDP realizations. As in (Younes and Simmons, 2006; Legay et al., 2010), we employ a sequential scheme that allows sampling only the number of paths necessary to ensure *a priori* probabilities α and β of type-1 errors (wrongly concluding that the property is false) and type-2 errors (wrongly concluding that it is true), respectively. Our approach builds on (Younes and Simmons, 2006; Legay et al., 2010) and extends it to handle reward and causal effect properties, by defining a suitable sequential test for T-distributed outcomes (rather than Bernoulli ones as done in Younes and Simmons (2006) and Legay et al. (2010)).

5.1. Computation of counterfactuals and causal effects

SMC relies on sampling paths of the (counterfactual) MDP model under some policy. We choose to sample these paths using the Gumbel-Max trick (see (5)) as it facilitates inference for the causal effect operator, as we will explain next. Using this formulation, we can express the counterfactual probability of (7) as the expectation of a function $f(\mathbf{G})$ of (prior) Gumbel variables $\mathbf{G} \sim$ Gumbel, as follows:

$$I_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau) = \mathbb{E}_{\mathbf{G}}[\mathrm{f}(\mathbf{G})], \text{with } \mathrm{f}(\mathbf{G}) = \mathbf{1}(\mathcal{P}',\pi',\tau'(\boldsymbol{G}),1) \models \phi), \quad (9)$$

where $\mathbf{1}$ is the indicator function, $\mathcal{P}' = \mathcal{P}^{\tau[|\tau|-t:]}$ is the counterfactual MDP, π' is intervention I's policy, and $\tau'(\mathbf{G})$ is the path of $\mathcal{P}'$ under π' which is uniquely determined by $\mathbf{G}$.[7]

The corresponding formulation for the counterfactual reward of (8) is readily obtained as:

$$R^{\leq k}_{=?}(\mathcal{P},\pi,\tau) = \mathbb{E}_{\mathbf{G}}[f(\mathbf{G})], \text{with } f(\mathbf{G}) = \sum_{\mathrm{i}=1}^{k} \mathcal{R}(\tau'(\mathbf{G})[\mathrm{i}]). \quad (10)$$

We proceed similarly for causal effect operators, with one important difference. While in Definition 7, we formulated the causal effect as the difference of two independent probabilities (or expected rewards), we here express it as the mean of paired differences between individual outcomes. This will allow us to reduce a two-sample inference problem into a one-sample problem.

$$\Delta^{I_1,I_0}_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau): \quad f(\mathbf{G}) = \mathbf{1}(\mathcal{P}',\pi_1,\tau_1(\mathbf{G}),1) \models \phi) - \mathbf{1}(\mathcal{P}',\pi_0,\tau_0(\mathbf{G}),1) \models \phi) \quad (11)$$

$$\Delta^{I_1,I_0}_{@t}.R^{\leq k}_{=?}(\mathcal{P},\pi,\tau): \quad f(\mathbf{G}) = \sum_{i=a}^{b} \mathcal{R}(\tau_1(\mathbf{G})[i]) - \mathcal{R}(\tau_0(\mathbf{G})[i]), \quad (12)$$

where for $i = 0, 1$, π_i is I_i's policy, and $\tau_i(\mathbf{G})$ is the path of the counterfactual MDP $\mathcal{P}'$ under π_i uniquely determined by the Gumbel $\mathbf{G}$. The advantage of the above form using paired differences is that this yields smaller variability, and hence, a more accurate statistical estimation, than the one based on the difference of independent means.

5.2. Qualitative properties

Let $p_\phi = I_{@t}.P_{=?}(\phi)(\mathcal{P},\pi,\tau)$ be the true (unknown) counterfactual probability of ϕ for a given MDP $\mathcal{P}$, policy π, and path τ. The problem of checking whether p_ϕ is above a given threshold θ, that is deciding property $I_{@t}.P_{\geq\theta}(\phi)$, can be formulated and solved as one of hypothesis testing, where we test the hypothesis $H: p_\phi \geq \theta$ against $K: p_\phi < \theta$ using a set of observations $x_1, \ldots, x_m$ of the underlying process.

Hypothesis testing may incur two kinds of errors: *type-1 errors*, that is wrongly concluding that K is true (when H holds) and *type-2 errors*, that is wrongly concluding that H is true (when K holds). We denote the probability of *type-1 errors* by α and that of *type-2 errors* by β. The pair $\langle\alpha,\beta\rangle$ is also called the *strength* of the test.

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

Wald's sequential probability ratio test (SPRT) (Wald, 2004) is an efficient scheme used in probabilistic model checking (Younes and Simmons, 2006; Younes et al., 2006) to sample only the number of realizations necessary to answer the above hypothesis test with strength $\langle \alpha, \beta \rangle$. We first explain in detail the SPRT for $I_{@t}.P_{\geq\theta}(\phi)$ properties, and then briefly cover the other kinds of formulas.

$I_{@t}.P_{\geq\theta}(\phi)$ properties. The SPRT method considers the following relaxation of the original hypotheses: $H_0 : p_\phi \geq \theta_0$ VS $H_1 : p_\phi \leq \theta_1$, with $\theta_0 = \theta + \delta$ and $\theta_1 = \theta - \delta$, where $\delta > 0$ is a user-defined parameter. The interval (θ_1,θ_0) is called *indifference region*, as we are willing to accept either hypothesis when $p_\phi \in (\theta_1,\theta_0)$. This relaxation is necessary because, when testing the original hypotheses H and K, we cannot control simultaneously both α and β if the true probability p_ϕ is exactly equal to θ, see (Younes and Simmons, 2006; Younes et al., 2006).

In the SPRT, we collect observations iteratively. At the m-th iteration, we have m observations $\mathbf{x}_m = (x_1, \ldots, x_m)$. In our case, these are counterfactual outcomes, that is realizations of the Bernoulli process $(X_1, \ldots, X_m)$ where $X_i \sim f(\mathbf{G}) = \mathbf{1}(\mathcal{P}', \pi', \tau'(\mathbf{G}), 1) \models \phi)$ (see Eq. 9). Given $\mathbf{x}_m$, we compute the following likelihood ratio (LR)

$$\frac{f(\mathbf{x}_m \mid H_1)}{f(\mathbf{x}_m \mid H_0)} = \frac{\prod_{i=1}^m Pr(X_i = x_i \mid p_\phi = \theta_1)}{\prod_{i=1}^m Pr(X_i = x_i \mid p_\phi = \theta_0)} = \frac{\theta_1^{d_m}(1-\theta_1)^{m-d_m}}{\theta_0^{d_m}(1-\theta_0)^{m-d_m}}, \tag{13}$$

where $d_m = \sum_{i=1}^m x_i$ is the number of observed successes. In other words, $f(\mathbf{x}_m \mid H_i)$ is the probability of observing the sequence $\mathbf{x}_m$ if $p_\phi = \theta_i$ holds. At this point, the SPRT compares the LR with the constants $A = (1-\beta)/\alpha$ and $B = \beta/(1-\alpha)$ and: if $\frac{f(\mathbf{x}_m|H_1)}{f(\mathbf{x}_m|H_0)} \leq B$, we accept H_0, with a type-2 error probability of β;

if $\frac{f(\mathbf{x}_m|H_1)}{f(\mathbf{x}_m|H_0)} \geq A$, we accept H_1, with a type-1 error probability of α; or,

we collect additional observations until one of the two above conditions hold. Note that this procedure requires a larger number of observations as the true p_ϕ approaches the threshold θ. Nevertheless, a decision is always reached after a finite number of steps (see Younes and Simmons (2006); Younes et al. (2006) for a more detailed analysis of the SPRT's stopping time). The above decision scheme is valid for other kinds of properties as well, that is it doesn't depend on the underlying distribution of the observations, as long as the LR is adequately defined. Hence, we won't repeat it for the cases below.

$I_{@t}.R^{\lesseqgtr\not\theta}_{\geq\theta}$ formulas. The SPRT can be also applied to variables other than Bernoulli, as are those entailed by reward-based properties. The corresponding test is an application of the SPRT to T-distributed observations (Schnuerch and Erdfelder, 2020). Let μ be the true (unknown) average cumulative reward, that is $\mu = I_{@t}.R^{\lesseqgtr\not\theta}_{=?}$. Here, we sample observations from the $f(\mathbf{G})$ of (10), for which we have that $\mu = \mathbb{E}[f(\mathbf{G})]$.

We consider the hypotheses: $H_0 : \mu \geq \theta_0$ VS $H_1 : \mu \leq \theta_1$, with $\theta_0 = \theta + \delta \cdot \sigma$ and $\theta_1 = \theta - \delta \cdot \sigma$, where σ is the (unknown) standard deviation of $f(\mathbf{G})$, and $\delta > 0$ is the indifference parameter: the indifference region spans $2 \cdot \delta$ standard deviations around θ.

The definition of the LR follows the intuition that if H_0 holds and in particular, $\mu = \theta_0$, then the variable $T_m = (\bar{X}_m - \theta)/S_m$ follows a non-central T distribution with non-centrality parameter $\delta \cdot \sqrt{m}$ and $m-1$ degrees of freedom, where $\bar{X}_m = \frac{1}{m}\sum_{i=1}^m X_i$ is the sample mean and $S_m = \frac{1}{\sqrt{m}}\sqrt{\frac{1}{m-1}\sum_{i=1}^m (X_i - \bar{X}_m)^2}$ is the standard error of $\bar{X}_m$ (Schnuerch and Erdfelder, 2020). The same reasoning holds for H_1, but after adjusting the sign of T_m. Hence, the LR is given by $\frac{f(\mathbf{x}_m|H_1)}{f(\mathbf{x}_m|H_0)} = \frac{f_T(-t_m|m-1,\delta\cdot\sqrt{m})}{f_T(t_m|m-1,\delta\cdot\sqrt{m})}$ where t_m is the observed value of T_m and $f_T(x \mid m-1, \delta \cdot \sqrt{m})$ is the p.d.f. at x of the non-central T distribution with $m-1$ degrees of freedom and parameter $\delta \cdot \sqrt{m}$.

$\Delta^{I_1,I_0}_{@t}.P_{\geq\theta}(\phi)$ and $\Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr\not\theta}_{\geq\theta}$ formulas. Since we can express the causal effect as the mean of a (non-Bernoulli) variable (the paired difference in the counterfactual outcomes of I_1 and I_0), we can apply the same SPRT procedure introduced above for $I_{@t}.R^{\lesseqgtr\not\theta}_{\geq\theta}$ formulas,[8] provided that we use the correct definition of $f(\mathbf{G})$, that is that of (11) for $\Delta^{I_1,I_0}_{@t}.P_{\geq\theta}(\phi)$ and (12) for $\Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr\not\theta}_{\geq\theta}$.

When $I_1 = I_0$, however, the above procedure fails because the two policies attain the same outcomes, and so their pairwise differences are constantly 0, resulting in $S_m = 0$ and $T_m = \infty$, which has a likelihood of 0. To detect this case, as done in David et al. (2011), we run a dedicated SPRT to test that the probability of obtaining equal outcomes is equal to 1.

Boolean combinations. To verify $\neg\Phi$ with strength $\langle \alpha, \beta \rangle$, we verify Φ with strength $\langle \beta, \alpha \rangle$ and negate the result. To verify a conjunction $\bigwedge_{i=1}^N \Phi_i$ with strength $\langle \alpha, \beta \rangle$, we need to verify each conjunct Φ_i with strength $\langle \alpha/N, \beta \rangle$. See (Younes and Simmons, 2006) for more details.

5.3. Quantitative properties

For quantitative properties, we use Chernoff-Hoeffding bounds to identify the number of realizations n necessary such that the Monte-Carlo estimate of the probability (or reward) property meets *a priori* error and confidence bounds. Given an error bound $\delta > 0$ and an iid sample $X_1, \ldots, X_n$ such that for each $i = 1, \ldots, n$, $\mathbb{E}[X_i] = \mu$ and $x_l \leq X_i \leq x_u$ for some constant $x_l < x_u$, then the Hoeffding inequality (Hoeffding, 1963) establishes that $P(|\bar{X}_n - \mu| \geq \delta) \leq 2\exp\left(-\frac{2n\delta^2}{(x_u - x_l)^2}\right)$, where $\bar{X}_n = (1/n)\sum_{i=1}^n X_i$ is the sample mean. Hence, given bounds $\delta > 0$ and $0 < \alpha < 1$, one can determine *a priori* the number of realizations n such that $P(|\bar{X}_n - \mu| \geq \delta) \leq \alpha$, by equating $\alpha = 2\exp\left(-\frac{2n\delta^2}{(x_u - x_l)^2}\right)$ and obtaining $n = \left\lceil -\frac{(x_u - x_l)^2 \log(\alpha/2)}{2\delta^2} \right\rceil$.

For the special case of $I_{@t}.P_{=?}(\phi)$ properties, $\bar{X}_n$ is the sample estimate of the probability, μ is the probability to estimate (and hence $0 < \delta < 1$), $x_l = 0$ and $x_u = 1$. For $\Delta^{I_1,I_0}_{@t}.P_{=?}(\phi)$ properties, we have that $x_l = -1$ and $x_u = 1$ as these are the ranges for the difference of two Bernoulli outcomes. For $I_{@t}.R^{\lesseqgtr\not?}_{=?}$ formulas, each realization is a cumulative reward value, hence $x_l = \not\ell \cdot \mathcal{R}_l$ and $x_u = \not\ell \cdot \mathcal{R}_u$ where $\mathcal{R}_u$ and $\mathcal{R}_l$ are, respectively, the largest and smallest values of the MDP's reward function $\mathcal{R}$. Hence, for $\Delta^{I_1,I_0}_{@t}.R^{\lesseqgtr\not?}_{=?}$ formulas, we have $x_u = \not\ell(\mathcal{R}_u - \mathcal{R}_l)$ and $x_l = \not\ell(\mathcal{R}_l - \mathcal{R}_u)$.

These *a priori* bounds, however, might be too conservative, especially for reward properties where the range $(x_u - x_l)$ tends to be consistently larger than what observed empirically. An alternative is to compute confidence intervals, that is fix the sample size n and the confidence $1 - \alpha$, thereby obtaining an estimate $\bar{X}_n$ and an interval $[\bar{X}_n]_\alpha \ni \bar{X}_n$ such that $P(\mu \notin [\bar{X}_n]_\alpha) = \alpha$. In this sense, the width of $[\bar{X}_n]_\alpha$ is comparable to the δ bound in Hoeffding

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

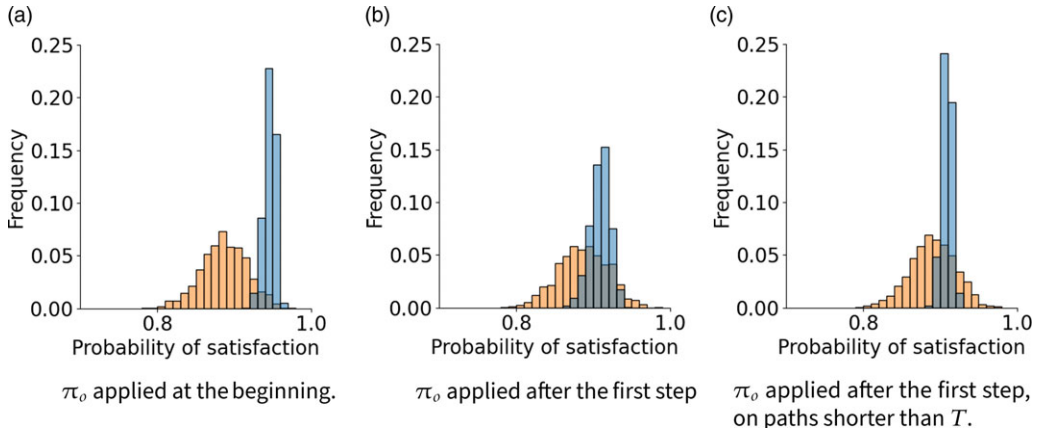


Figure 5. Counterfactual probabilities under the optimal policy π_o (blue) given that we observe MDP paths under the nominal policy π (orange). In (a) and (b) paths have length 10 (same as the time bound T in ϕ). In (c), we observe paths of length $2 < T$, and so, applying π_o results in paths that are part counterfactual, part post-interventional.

inequality. To construct confidence intervals for $I_{@t}.P_{=?}(\phi)$ properties, one can use the common normal-approximation (aka Wald) interval if n is not too small or the true probability not too close to 0 or 1,[9] or use the 'exact" (but usually conservative) Clopper-Pearson interval. For the other properties, we can construct one-sample mean intervals using the T distribution.

5.4. Algorithmic complexity

The complexity of SMC is $O(\mathscr{k}{\cdot}N{\cdot}c_\phi)$ where $\mathscr{k}$ is the number of counterfactual operators in the formula, N is the number of sampled paths (for each operator), and c_ϕ is the cost of evaluating the operator's path formula ϕ on each path. The latter term has complexity $O((2|\tau|)^d)$ where $|\tau|$ is the path length (bounded by the temporal bounds in ϕ) and d is the depth of ϕ, that is the maximum number of nested until expressions (Bartocci et al., 2018). The term N is a random variable (owing to the randomness of the sample) and its expected value depends on the true (unknown) probability p to evaluate and the error bounds α and β. Formulas for $\mathbb{E}[N]$ can be found for specific values of p, see (Younes and Simmons, 2006), but no general analytical form exists. Nevertheless, the SMC algorithm terminates with probability 1 (Younes and Simmons, 2006).

6. Experimental evaluation

We provide two sets of results. In the first one, we consider a simple grid-world model and a reach-avoid specification. We use this case study to provide a detailed analysis of interventional and counterfactual probabilities, their variability, and the accuracy of counterfactual inference. In the second set of results, we use PCFTL on four complex 2D grid-world environments from the MiniGrid library (Chevalier-Boisvert et al., 2018). Although these two case studies evaluate our approach on relatively similar GridWorld MDPs, we can still explore a wide range of logical specifications and properties within these environments.

6.1. Reach-avoid example

We consider a 4×4 grid-world example, where the agent can move up, down, left, or right, one square at a time. The specification ϕ is one of reach-avoid: we want to reach some goal region while avoiding an unsafe region, that is $\phi \equiv \neg\text{unsafe}\ \mathcal{U}_{[0,T]}\text{goal}$. We choose $T = 10$. We consider two policies, a nominal (default) policy π and an optimal policy π_o. The optimal policy is found by value iteration after assigning a reward 1 to the goal and making the unsafe and goal states terminal. The nominal policy is defined manually to make it intentionally less safe than π_o. The stochasticity comes from the fact that the environment, with small probability (0.1 in our experiments), randomly takes the agent to a different position than that determined by the policy.

For each experiment in this subsection, we perform 1000 repetitions to evaluate the variability of the estimates. For each repetition, we generate 100 observed paths under the nominal policy. Counterfactuals are estimated using 20 posterior Gumbel realizations. Probability values are computed by averaging the satisfaction value of ϕ over all paths within each repetition. We choose the optimal policy as the interventional/counterfactual one, by defining $I = \{\pi \leftarrow \pi_o\}$.

We evaluate the performance of the optimal policy in a counterfactual setting. In particular, we compare the probability $P_{=?}(\phi)$ under the nominal MDP against the average counterfactual probability $I_{@t}.P_{=?}(\phi)$ for some t (where the average is w.r.t. the set of nominal paths used for $P_{=?}(\phi)$). We apply I at the beginning of the path ($t = |\tau| - 1$, see Figure 5a) and after the first step ($t = |\tau| - 2$, see Figure 5b). Since π_o (blue histograms in Figure 5) is safer than π (orange), the distribution of counterfactual probabilities clearly dominates that under nominal settings. See Figure 5a. For the same reason, delaying the intervention of one step leads to more unsafe trajectories (the blue histogram in Figure 5b is indeed shifted to the left compared to that in Figure 5a).

In Figure 5a, we provide results of a query corresponding to the scenario of Figure 4c, that is involving both the counterfactual past and the subsequent future evolution of the system. To do so, we draw paths τ under π of length 2 (shorter than than ϕ's time bound) and apply I after the first time step. This results in paths that are counterfactual in the first part (because we apply I in the past, conditioned by τ) and post-interventional in the second part (because to evaluate ϕ, we need paths longer than the observed τ).

6.2. MiniGrid benchmark

MiniGrid (Chevalier-Boisvert et al., 2018) is a collection of 2D grid-world environments with goal-oriented tasks designed for developing reinforcement learning algorithms. Each cell in this grid world is encoded as a three-dimensional tuple (object, color, state). There are 8 different objects, 6 colors and 3 states: open, close and locked. There are 7 actions that the agent can take which are turn left, turn right, move forward, pick up, drop, toggle and

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

Table 1. PCFTL verification of the MiniGrid benchmark, with 6 x 6 grids. For each environment, we apply the intervention at the start of the path ($t = T - 1$) and 10 steps after the start ($t = T - 11$). $T = 50$ is the length of the path. The SMC parameters (see Section 5) are $\delta = 0.02$, and $\alpha = 0.05$ and $\beta = 0.2$ for P and $I_{@t}.P$ properties, and $\alpha = 0.01$ and $\beta = 0.2$ for $\Delta^{I,\emptyset}_{@t}.P$. $\top$ and $\bot$ indicate whether the SMC procedure returns true or false for the given PCFTL formulae, and in parentheses are the number of realizations required by SMC to reach this verdict

Environment	t	$P_{\geq 0.9}(\phi)$	$I_{@t}.P_{\geq 0.9}(\phi)$	$\Delta^{I,\emptyset}_{@t}.P_{>0}(\phi)$
DoorKey6x6	$T-1$	$\top$(125)	$\bot$(25)	$\bot$(50)
DoorKey6x6	$T-11$	$\top$(125)	$\bot$(25)	$\bot$(50)
Empty6x6	$T-1$	$\top$(75)	$\bot$(25)	$\bot$(50)
Empty6x6	$T-11$	$\top$(75)	$\bot$(25)	$\bot$(75)
Fetch6x6	$T-1$	$\top$(75)	$\bot$(25)	$\bot$(75)
Fetch6x6	$T-11$	$\top$(75)	$\bot$(50)	$\bot$(75)
GoToDoor6x6	$T-1$	$\top$(125)	$\bot$(25)	$\bot$(50)
GoToDoor6x6	$T-11$	$\top$(125)	$\bot$(25)	$\bot$(100)

done. We consider four of these environments: *Empty*, *DoorKey*, *GoToDoor* and *Fetch*.

Empty is the simplest environment, where the agent simply navigates the grid to reach some goal. This corresponds to the specification $\mathcal{F}_{[1,T]}$goal, where we choose $T = 50$. In the *DoorKey* environment, a key and a door exist on the grid. The agent must first find the key, unlock the door, and reach the goal, expressed as $\phi \equiv \mathcal{F}_{[1,T_k]}(\text{key} \wedge \mathcal{F}_{[1,T_d]}(\text{door} \wedge \mathcal{F}_{[1,T_g]}\text{goal}))$, with $T_k + T_d + T_g = T$. This task requires the agent to learn basic navigation skills and non-trivial sequential plans. In *GoToDoor*, we have four doors with different colors, and the agent is tasked to reach the door of some given color: $\phi \equiv \mathcal{F}_{[1,T]}$door. The door is always unlocked, making it a simpler task than *DoorKey*. In *Fetch*, the grid contains multiple objects with assorted colors which the agent must pick up and bring to the goal: $\phi \equiv \mathcal{F}_{[1,T_o]}(\text{object} \wedge \mathcal{X}(\text{carrying}\,\mathcal{U}_{[1,T_g]}\text{goal}))$, with $T_o + T_g + 1 = T$. This task requires learning to manipulate objects and navigate the grid.

For each environment, we train two convolutional neural network policies using the Proximal Policy Optimization (PPO) (Schulman et al., 2017) algorithm. For the nominal policy π, this time we use an optimal policy, trained using 10 million time steps. The interventional/counterfactual policy is intentionally undertrained, using only 200 time steps.

Experimental results are presented in Table 1. We examine probabilities under the nominal/optimal policy (3rd column) using the formula $P_{\geq 0.9}(\phi)$, counterfactual probabilities with the undertrained policy (4th column) using $I_{@t}.P_{\geq 0.9}(\phi)$, and determine the causal effect between the two (5th column) using $\Delta_{@t}{}^{I,\emptyset}.P_{>0}(\phi)$. For every environment and PCFTL formula, we carry out two set of experiments using two different intervention points, at the start of the trajectory ($t = T - 1$), and 10 steps into the trajectory ($t = T - 11$).

Verification results are computed using statistical model checking (see 5 for details on the decision procedures). Results indicate that the system does not satisfy the property when using an undertrained policy, while the optimal policy is always successful. This performance gap can also be seen in the causal effect column. We observe that the verification procedure is very efficient (requiring at most 125 realizations), and that the number of realizations necessary to obtain a positive answer for the nominal policy is higher than those for a negative answer for the interventional policy. The reason is that we set a high probability threshold, $p \geq 0.9$, and so, even with a well-trained policy, we require a fair amount of evidence to conclude that the property is satisfied. Conversely, the interventional policy performs enough badly to require much fewer points for concluding that the property is violated.

7. Related work

Causality and verification. Concepts of causality have been investigated in formal verification for years (Baier et al., 2021). Two main classes of approaches exist, respectively based on the theory of actual causality (Halpern, 2016; Halpern and Pearl, 2020) and on probabilistic causation. Given an SCM $\mathcal{M}$ and a context u, an actual cause is, informally, the smallest set of SCM variables that, if forced with a different value, lead to a different (counterfactual) outcome for some target variable Y. This notion has been adapted in Beer et al. (2012) to find so-called root causes in LTL counterexample traces, in Gössler and Aştefănoaei (2014) to identify the components of a timed-automata network responsible for a given failure trace, and in Leitner-Fischer and Leue (2013) to derive fault trees from probabilistic counterexamples. We note there also exist techniques not based on actual causality for finding root causes of failure in temporal logic monitoring (e.g., Bartocci et al., 2018; Zhang et al., 2023). Probabilistic causation methods like (Baier et al., 2021; Kleinberg and Mishra, 2009; Kleinberg, 2011; Baier et al., 2022) build on the probability-raising (PR) principle by which the probability of an effect E is higher after observing a cause C than if the cause had not happened. More precisely, these works consider Markov models and express E and C as sets of states or PCTL state formulas. Our work complements these methods as it focuses not on identifying causes given some observations but on reasoning about the probability of a temporal logic specification in interventional and counterfactual settings. Methods based on actual causality similarly rely on counterfactuals but consider only non-probabilistic models. Methods based on the PR principle support probabilistic models but do not support counterfactual analysis.

Two relevant papers have been published in the last year at the intersection between causality and temporal logic (TL). In Coenen et al. (2022), the authors extend actual causality to the case where causes and effects are given as TL properties. Their work is different from ours in that they do not consider probabilistic systems, plus they use TL to specify causes and effects, but the logic itself cannot express counterfactual queries. The work closest to ours is (Finkbeiner and Siber, 2023), which introduces a new (non-probabilistic) counterfactual TL with *would* and *might* modalities, borrowed from Lewis' theory of counterfactuals (Lewis, 2013). However, their method is model-agnostic, that is counterfactuals are obtained by manipulating the observed trace, regardless of the model that generated it. Our counterfactual traces are instead obtained by intervening on the data-generating model.

Probabilistic hyperproperties. Probabilistic hyper-properties (PHPs) for MDPs have been recently introduced in (Dimitrova et al., 2020; Ábrahám et al., 2020) to support quantification over MDP schedulers (i.e., policies). One can see that PHPs for MDPs are strictly more expressive than the fragment of PCFTL without counterfactuals (i.e., where interventions can be applied only at $t = 0$). For instance, the PCFTL causal-effect formula $\Delta^{I_1,I_0}_{@t}.P_{\bowtie p}(\phi)$

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

can be expressed as the PHP $\exists\sigma_1\exists\sigma_0.P(\phi_{\sigma_1}) - P(\phi_{\sigma_0}) \bowtie p$ (using the syntax of (Dimitrova et al., 2020)) where the domains of schedulers σ_0 and σ_1 are singletons (and chosen to be consistent with I_0 and I_1, respectively). However, PHPs do not support counterfactuals, which is the main strength of our method.

Causality in reinforcement learning. There is a growing interest in applying causal inference in RL, for instance, to evaluate counterfactual policies from observational data (Oberst and Sontag, 2019), provide counterfactual explanations (Tsirtsis, De, and Rodriguez, 2021) (i.e., the minimum number of policy actions to change in order to attain a better outcome), produce counterfactual data to enhance training of RL policies (Forney et al., 2017; Buesing et al., 2018), or estimate causal effects in presence of confounding factors (Lu, Schölkopf, and Hernández-Lobato, 2018). These works are very relevant yet they consider different problems from ours. That said, PCFTL builds on (Oberst and Sontag, 2019) which introduces Gumbel-Max SCMs and their counterfactual stability. More recently, other methods have shown that the Gumbel-Max SCM is not the only causal model that satisfies the counterfactual stability property, and instead bound over all models that satisfy counterfactual stability (Haugh and Singal, 2023), or search for a particular model that optimises some given criteria (Lorberbom et al., 2021). In this paper, we limit our attention to only Gumbel-Max SCMs since other methods are either computationally inefficient or require extra assumptions.

8. Conclusion

We have presented the probabilistic temporal logic PCFTL, the first of its kind to enable causal reasoning about interventions, counterfactuals, and causal effects in Markov Decision Processes. From a syntactic viewpoint, this is achieved by introducing an operator that subsumes interventions, counterfactuals, and the traditional probabilistic operator. The semantics of PCFTL makes use of counterfactual MDPs constructed from Gumbel-Max structural causal models, which provide a representation of discrete-state MDPs amenable to counterfactual reasoning. We performed a set of experiments on a benchmark of grid-world models, demonstrating the usefulness of the approach (being applicable to deep reinforcement learning policies as well) and the accuracy of counterfactual inference. We envision several future directions for this work, including investigating numerical or symbolic (as opposed to statistical) model-checking algorithms, and extending our approach to a broader range of systems, such as uncertain, partially observable, and continuous-time and continuous-state MDPs. Achieving this will require developing robust counterfactual inference methods tailored to these different complex systems but will ultimately enable PCFTL to be applied more broadly across a diverse set of cyber-physical and data-driven systems.

Data availability. The code for running the experiments is publicly available on Zenodo at https://doi.org/10.5281/zenodo.10619287.

Credit authorship contribution. **Milad Kazemi:** Formal analysis, Investigation, Methodology, Software, Validation, Visualizations, Writing. **Jessica Lally:** Formal analysis, Methodology, Visualizations, Writing. **Nicola Paoletti:** Conceptualization, Formal analysis, Funding acquisition, Methodology, Resources, Supervision, Writing.

Financial support. This work was supported by UK Research and Innovation [grant number EP/W014785/2]; and UK Research and Innovation [grant number EP/S023356/1] in the UKRI Centre for Doctoral Training in Safe and Trusted Artificial Intelligence (www.safeandtrustedai.org).

Competing interests. None.

Ethics statement. Ethical approval and consent are not relevant to this paper.

Connections references

Paoletti N and Woodcock J (2023) How to ensure safety of learning-enabled cyber-physical systems? *Research Directions: Cyber-Physical Systems* **1**, e2, 1–2. https://doi.org/10.1017/cbp.2023.2.

Notes

1 As we will explain in Section 2.3, there could be multiple such what-if paths, and so the counterfactual probability of ϕ could take other values than 1 or 0.
2 $P_{\mathcal{M}[X\leftarrow x]}(Y)$ is often written as $P(Y|do(X=x))$ in Pearl's *do* notation.
3 Note that $P_{\mathcal{M}}(Y|X=x)$ is, in general, different from the desired $P_{\mathcal{M}[X\leftarrow x]}(Y)$ because conditioning on $X=x$ alone doesn't prevent unwanted spurious associations.
4 In contrast, the approach based on the inverse CDF trick, where $f(S_t, A_t, U_t)$ is the U_t-quantile of $P_{\mathcal{P}}S_{t+1} \mid S_t, A_t)$ and $U_t \sim \text{Unif}(0,1)$, does not enjoy counterfactual stability and is highly sensitive to permutations of the state ordering (note that imposing some ordering is required by the quantile function).
5 In particular, to determine the satisfaction of path formula $(\mathcal{P}, \pi, \tau, t) \vDash \Phi$, we evaluate state formula Φ over the path prefix $\tau[1:t]$ (to allow for potentially nested counterfactual operators), while in PCTL*, Φ would be evaluated over state $\tau[t]$.
6 For an MDP with state space $\mathcal{S}$ and transition probabilities P, the definition of the augmented MDP is trivial: it will have state space $\bigcup_{1\le i\le T}\mathcal{S}^i$ and transition probabilities $P(s'_1 \dots s'_T \mid s_1 \dots s_T, a) = P(s'_T \mid s_T, a)$ if $\bigwedge_{t=2}^{T} s'_{t-1} = s_t$ and 0 otherwise.
7 This decomposition is analogous to how we obtain the distribution $P_{\mathcal{M}}$ of an SCM $\mathcal{M}$ as a function of the distribution $P(\mathbf{U})$ of its exogenous variables $\mathbf{U}$.
8 For the special case of $\Delta^{I_1,I_0}_{@t}.P_{>0}(\phi)$, an alternative sequential test could be used, see (David et al., 2011).
9 For the normal approximation to be valid, we require $n \cdot p_\phi \ge 10$ and $n \cdot (1-p_\phi) \ge 10$.

References

Baier C (1998) On algorithmic verification methods for probabilistic systems. PhD diss., Habilitation thesis, Fakultät für Mathematik & Informatik, Universität Mannheim.

Baier C, Clarke EM, Hartonas-Garmhausen V, Kwiatkowska M and Ryan M (1997) Symbolic model checking for probabilistic processes. In *Automata, Languages and Programming*. Berlin, Heidelberg: Springer, 430–440.

Baier C, Dubslaff C, Funke F, Jantsch S, Majumdar R, Piribauer J and Ziemek R (2021) From verification to causality-based explications (invited talk). *Leibniz International Proceedings in Informatics (LIPIcs)* **198**, 1–20.

Baier C, Funke F, Jantsch S, Piribauer J and Ziemek R (2021) Probabilistic causes in Markov chains. *International Symposium on Automated Technology for Verification and Analysis* **18**, 347–367.

Baier C, Funke F, Piribauer J and Ziemek R (2022) On probability-raising causality in markov decision processes. *Fossacs* **13242**, 40–60.

Baier C and Katoen P-J (2008) *Principles of Model Checking*. Cambridge: MIT Press.

Bartocci E, Deshmukh J, Donzé A, Fainekos G, Maler O, Nickovic D and Sankaranarayanan S (2018) Specification-based monitoring of cyber-physical systems: a survey on theory, tools and applications. In *Lectures on Runtime Verification*. Cham: Springer, 135–175.

Bartocci E, Ferrère T, Manjunath N and Nickovic D (2018) Localizing faults in simulink/stateflow models with STL. In *Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (Part of CPS Week)*. New York: Association for Computing Machinery.

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.

Beer I, Ben-David S, Chockler H, Orni A and Trefler R (2012) Explaining counterexamples using causality. *Formal Methods in System Design* **40**(1), 20–40.

Buesing L, Weber T, Zwols Y, Heess N, Racaniere S, Guez A and Lespiau B-J. Woulda, coulda, shoulda: counterfactually-guided policy search. *International Conference on Learning Representations*, published online 15 November 2018, doi: 10.48550/arXiv.1811.06272.

Chevalier-Boisvert M, Willems L and Pal S (2018) *Minimalistic gridworld environment for Gymnasium*. https://github.com/Farama-Foundation/Minigrid.

Coenen N, Finkbeiner B, Frenkel H, Hahn C, Metzger N and Siber J (2022) Temporal causality in reactive systems. In *International Symposium on Automated Technology for Verification and Analysis*. Berlin, Heidelberg: Springer, 208–224.

David A, Larsen KG, Legay A, Mikucionis M, Poulsen BD, Van Vliet J and Wang Z (2011) Statistical model checking for networks of priced timed automata. *International Conference on Formal Modeling and Analysis of Timed Systems* **6919**, 80–96.

Dimitrova R, Finkbeiner B and Torfah H (2020) Probabilistic hyperproperties of Markov decision processes. *International Symposium on Automated Technology for Verification and Analysis* **12302**, 484–500.

Finkbeiner B and Siber J (2023) Counterfactuals modulo temporal logics. arXiv preprint arXiv:2306.08916. Available at https://doi.org/10.48550/arXiv.2306.08916 (accessed 15 June, 2023).

Forney A, Pearl J and Bareinboim E (2017) Counterfactual data-fusion for online reinforcement learners. *International Conference on Machine Learning* **70**, 1156–1164.

Glymour M, Pearl J and Jewell NP (2016) *Causal Inference in Statistics: A Primer*. New York, NY: John Wiley & Sons.

Gössler G and Astefanoaei L (2014) Blaming in component-based real-time systems. *Proceedings of the 14th International Conference on Embedded Software* **7**, 1–10.

Guo R, Cheng L, Li J, Hahn PR and Liu H (2020) A survey of learning causality with data: problems and methods. *ACM Computing Surveys (CSUR)* **53**(4), 1–37.

Halpern JY (2016) *Actual Causality*. Cambridge: MIT Press.

Halpern JY and Pearl J (2020) Causes and explanations: a structural-model approach. Part i: causes. *The British Journal for the Philosophy of Science* **56**, 843–887.

Haugh MB and Singal R. Bounding counterfactuals in hidden markov models and beyond. *Available at SSRN 4529724*, published online 8 August 2023, doi: 10.2139/ssrn.4529724.

Hoeffding W (1963) Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association* **58**(301), 13–30.

Kleinberg S (2011) A logic for causal inference in time series with discrete and continuous variables. *Twenty-Second International Joint Conference on Artificial Intelligence* **2**, 943–950.

Kleinberg S and Mishra B (2009) The temporal logic of causal structures. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. Pittsburgh, PA: AUAI Press.

Kwiatkowska M, Norman G and Parker D (2007) Stochastic model checking. In *International School on Formal Methods for the Design of Computer, Communication and Software Systems*. Berlin, Heidelberg: Springer, 220–270.

Lecarpentier E and Rachelson E (2019) Non-stationary markov decision processes, a worst-case approach using model-based reinforcement learning. In Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E and Garnett R (eds), *Advances in Neural Information Processing Systems* Vol. **32**. New York: Curran Associates, Inc.

Legay A, Delahaye B and Bensalem S (2010) Statistical model checking: an overview. In *International Conference on Runtime Verification*. Berlin, Heidelberg: Springer, 122–135.

Leitner-Fischer F and Leue S (2013) Probabilistic fault tree synthesis using causality computation. *International Journal of Critical Computer-Based Systems* **4**(2), 119–143.

Lewis D (2013) *Counterfactuals*. New York, NY: JohnWiley & Sons.

Lorberbom G, Johnson DD, Maddison CJ, Tarlow D and Hazan T (2021) Learning generalized gumbel-max causal mechanisms. *Advances in Neural Information Processing Systems* **34**, 26792–26803.

Lu C, Schölkopf B and Hernández-Lobato MJ (2018) Deconfounding reinforcement learning in observational settings. arXiv Preprint arXiv:1812.10576. Available at https://doi.org/10.48550/arXiv.1812.10576 (accessed 26 December, 2018).

Manna Z and Pnueli A (2012) *The Temporal Logic of Reactive and Concurrent Systems: Specification*. New York City: Springer Science & Business Media.

Oberst M and Sontag D (2019) Counterfactual off-policy evaluation with Gumbel-max structural causal models. *Proceedings of the 36th International Conference on Machine Learning, PMLR* **97**, 4881–4890.

Pearl J (2009) *Causality*, 2nd edn. Cambridge: Cambridge University Press.

Puterman ML (2014) *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY: John Wiley & Sons.

Schnuerch M and Erdfelder E (2020) Controlling decision errors with minimal costs: the sequential probability ratio t test. *Psychological Methods* **25**(2), 206.

Schulman J, Wolski F, Dhariwal P, Radford A and Klimov O (2017) Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347. Available at https://doi.org/10.48550/arXiv.1707.06347 (accessed 28 August, 2017).

Tsirtsis S, De A and Rodriguez M (2021) Counterfactual explanations in sequential decision making under uncertainty. *Advances in Neural Information Processing Systems* **34**, 30127–30139.

Wald A (2004) *Sequential Analysis*. Chelmsford, MA: Courier Corporation.

Younes HLS, Kwiatkowska M, Norman G and Parker D (2006) Numerical vs. statistical probabilistic model checking. *International Journal on Software Tools for Technology Transfer* **8**(3), 216–228.

Younes HLS and Simmons RG (2006) Statistical probabilistic model checking with a focus on time-bounded properties. *Information and Computation* **204**(9), 1368–1409.

Zhang Z, An J, Arcaini P and Hasuo I (2023) Online causation monitoring of signal temporal logic. *Computer Aided Verification - 35th International Conference, CAV 2023, Proceedings, Part I* **13964**, 62–84.

Zuliani P, Platzer A and Clarke EM (2013) Bayesian statistical model checking with application to stateflow/simulink verification. *Formal Methods in System Design* **43**, 338–367.

Downloaded from https://www.cambridge.org/core. 15 Nov 2026 at 21:31:52, subject to the Cambridge Core terms of use.