TECHNICAL REPORT
R-105
May, 2024

Eligibility Traces for Confounding Robust Off-Policy Evaluation

Junzhe Zhang and **Elias Bareinboim**
Causal Artificial Intelligence Lab
Columbia University
{junzhez,eb}@cs.columbia.edu

Abstract

A unifying theme in Artificial Intelligence is learning an effective policy to control an agent in an unknown environment in order to optimize a certain performance measure. Off-policy methods can significantly improve the sample efficiency during training since they allow an agent to learn from observed trajectories generated by different *behavior policies*, without directly deploying the *target policies* in the underlying environment. This paper studies off-policy evaluation from biased offline data where (1) *unobserved confounding* bias cannot be ruled out a priori; or (2) the observed trajectories do not *overlap* with intended behaviors of the learner, i.e., the target and behavior policies do not share a common support. Specifically, we first extend the Bellman's equation to derive effective closed-form bounds over value functions from the observational distribution contaminated with unobserved confounding and no-overlap. Second, we propose two novel algorithms that use eligibility traces to estimate these bounds from finite observational data. Compared to other partial identification methods for off-policy evaluation in sequential environments, these methods are model-free and do not rely on additional parametric knowledge about the system dynamics in the underlying environment.

1 Introduction

A typical reinforcement learning agent learns from past data, i.e., from observed trajectories of states, actions, and reward signals generated by the agent intervening in the underlying environment. This data reflects the influence of the decision-making policy used to allocate actions based on the observed state, which is called the *behavior policy*. This policy can be selected by the agent in the past or by a different demonstrator operating in the same environment. *Policy evaluation* studies the problem of evaluating the effectiveness of a candidate *target policy* from the combination of past data and theoretical assumptions about the environment. When the behavior and target policies coincide, the evaluation is called *on-policy* learning, in which the expected return of candidate policies given the agent's starting state (i.e., the value function) could be directly estimated with empirical means [34]. In practice, however, the learner might have to learn about policies different from the currently deployed one that generated the data, leading to the well-known problem of *off-policy* learning.

Off-policy learning is a popular area of research in reinforcement learning, as it allows for more efficient learning by using data from different agents and policies. Several algorithms have been proposed for off-policy evaluation from finite observations, including Q-learning [38, 37], importance sampling [35, 12], and temporal difference [28, 21]. These algorithms rely on two critical assumptions about the behavior policy. First, no unobserved confounder affects the behavior policy's selected action and the subsequent state and reward. Second, the behavior policy is stochastic, covering all intended actions the target policy selects given all observed states. When either of these assumptions does not hold, the effect of the target policy is generally not *identifiable*, i.e., the model assumptions are insufficient to uniquely determine the value function from the offline data [24, 39].

In recent times, researchers have been using partial identification methods to obtain reliable off-policy evaluation in situations where there are unobserved confounders, and the behavior and target policies have no common support [14, 39, 15, 22, 17, 7, 16]. Partial identification is a well-studied problem in causal inference [2, 40], econometrics [10, 27, 30, 32, 8, 36, 19], and dynamical systems [1, 23, 9, 20, 5]. It enables the derivation of informative bounds on target effects from confounded observational data. Despite these developments, researchers often must employ a combination of approaches and constraints. These include: (1) additional parametric assumptions about the system dynamics (i.e., reward function, transition distribution, and behavior policy) are invoked under which bounds are derived [14, 15, 22, 17, 7, 16]; (2) a model-based algorithm is applied, which requires estimation of the underlying system dynamics [39]; (3) the decision horizon is finite, i.e., the agent only determines a finite number of actions [14, 39, 22, 17, 16].

This paper studies model-free algorithms for robust off-policy evaluation from confounded offline data generated by behavior policy with no-overlap support. We propose novel partial identification algorithms using eligibility traces to obtain informative bounds over the expected return of candidate policies from offline data generated from an unknown Markov decision process with an infinite horizon. More specifically, our contributions are summarized as follows. (1) We extend the Bellman equation that permits one to derive optimal bounds over target value functions from the observational distribution generated by an unknown behavior policy. (2) We propose a causal off-policy temporal difference algorithm (C-TD(λ)) using eligibility traces to estimate bounds over the state value function from finite observations contaminated with unobserved confounding and no-overlap. (3) We introduce an alternative eligibility traces algorithm following tree backup (C-TB(λ)) that obtains bounds over the state-action value function from confounded observations. Finally, we evaluate our proposed algorithms using extensive simulations in synthetic environments. Due to space constraints, all proofs are provided in Appendix A; details on the experiment setup are provided in Appendix B.

1.1 Preliminaries

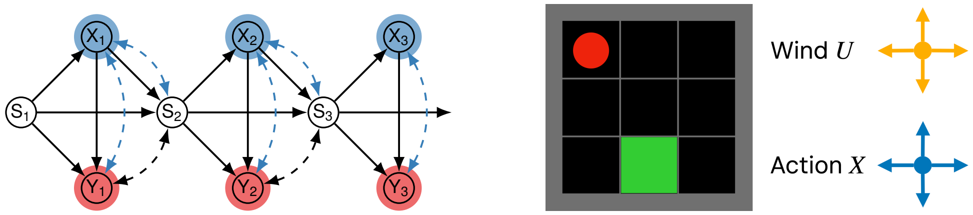
In this section, we introduce the basic notations and definitions used throughout the paper. We use capital letters to denote random variables (X), small letters for their values (x) and $\mathscr{D}_X$ for the domain of X. For an arbitrary set $\boldsymbol{X}$, let $|\boldsymbol{X}|$ be its cardinality. Fix indices $i, j \in \mathbb{N}$. Let $\boldsymbol{X}_{i:j}$ stand for a sequence of variables $\{X_i, X_{i+1}, \dots, X_j\}$; $\boldsymbol{X}_{i:j} = \emptyset$ if $j < i$. We denote by $P(\boldsymbol{X})$ represents a probability distribution over variables $\boldsymbol{X}$. Similarly, $P(\boldsymbol{Y} \mid \boldsymbol{X})$ represents a set of conditional distributions $P(\boldsymbol{Y} \mid \boldsymbol{X} = \boldsymbol{x})$ for all realizations $\boldsymbol{x}$. We will consistently use $P(\boldsymbol{x})$ as abbreviations for probabilities $P(\boldsymbol{X} = \boldsymbol{x})$; so does $P(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{X} = \boldsymbol{x}) = P(\boldsymbol{y} \mid \boldsymbol{x})$. Finally, $\mathbb{1}_{\boldsymbol{Z}=\boldsymbol{z}}$ is an indicator function that returns 1 if event $\boldsymbol{Z} = \boldsymbol{z}$ holds true; otherwise, it returns 0.

The basic semantical framework of our analysis rests on *structural causal models* (SCMs) [24, 3]. An SCM M is a tuple $\langle \boldsymbol{V}, \boldsymbol{U}, \mathcal{F}, P(\boldsymbol{U}) \rangle$, where $\boldsymbol{V}$ is a set of endogenous variables and $\boldsymbol{U}$ is a set of exogenous variables. $\mathcal{F}$ is a set of functions s.t. each $f_V \in \mathcal{F}$ decides values of an endogenous variable $V \in \boldsymbol{V}$ taking as argument a combination of other variables in the system. That is, $V \leftarrow f_V(\boldsymbol{PA}_V, \boldsymbol{U}_V), \boldsymbol{PA}_V \subseteq \boldsymbol{V}, \boldsymbol{U}_V \subseteq \boldsymbol{U}$. Exogenous variables $U \in \boldsymbol{U}$ are mutually independent, values of which are drawn from the exogenous distribution $P(\boldsymbol{U})$. Naturally, M induces a joint distribution $P(\boldsymbol{V})$ over endogenous variables $\boldsymbol{V}$, called the *observational distribution*.

An intervention on a subset $\boldsymbol{X} \subseteq \boldsymbol{V}$, denoted by $\mathrm{do}(\boldsymbol{x})$, is an operation where values of $\boldsymbol{X}$ are set to constants $\boldsymbol{x}$, replacing the functions $\{f_X : \forall X \in \boldsymbol{X}\}$ that would normally determine their values. For an SCM M, let $M_{\boldsymbol{x}}$ be a submodel of M induced by intervention $\mathrm{do}(\boldsymbol{x})$. For a set $\boldsymbol{Y} \subseteq \boldsymbol{V}$, the interventional distribution $P_{\boldsymbol{x}}(\boldsymbol{Y})$ induced by $\mathrm{do}(\boldsymbol{x})$ is defined as the joint distribution over $\boldsymbol{Y}$ in the submodel $M_{\boldsymbol{x}}$, i.e., $P_{\boldsymbol{x}}(\boldsymbol{Y}; M) \triangleq P(\boldsymbol{Y}; M_{\boldsymbol{x}})$. We leave M implicit when it is obvious from the context. For a detailed survey on SCMs, we refer readers to [24, Ch. 7] and [3].

2 Challenges of Causal Inconsistency

We will focus on the policy evaluation problem of an agent operating in a Markov Decision Process (MDP) [29] over a series of interventions $t = 1, 2, \dots$. For every time step t, the agent observes the current state S_t, performs an action $\mathrm{do}(X_t)$, receives a subsequent reward Y_t, and moves to the next state S_{t+1}. Values of the action X_t are selected by sampling from a stationary policy $\pi(x \mid s)$, which is a function mapping from the domain of the observed state S_t to the probability space over the domain of action X_t. Let $\boldsymbol{U}_t$ be an unobserved noise independently drawn from an exogenous



(a) MDP

(b) Windy Gridworld

Figure 1: (a) Causal diagram representing the data-generating mechanisms in a Markov Decision Process (MDP); (b) A windy gridworld environment where the red dot represents the agent and green square is the goal state; the agent can take five actions - `up`, `down`, `right`, `left`, and `stay-put`; the wind can blow in five direction - `north`, `south`, `west`, `east`, and `no-wind`.

distribution $P(\boldsymbol{U})$. Values of the subsequent reward Y_t and the next state S_{t+1} are, respectively, determined by structural functions $y_t \leftarrow f_Y(s_t, x_t, \boldsymbol{u}_t)$ and $s_{t+1} \leftarrow f_S(s_t, x_t, \boldsymbol{u}_t)$, taking as input the current state S_t, action X_t, and latent noise $\boldsymbol{U}_t$; values of S_1 are drawn from an initial distribution $P(S_1)$. We will consistently use $\mathcal{X}$, $\mathcal{S}$, and $\mathcal{Y}$ to denote the domain of every action X_t, state S_t, and reward Y_t. Like a standard discrete MDP, domains of actions $\mathcal{X}$ and states $\mathcal{S}$ are assumed to be finite; rewards are bounded in a real interval $\mathcal{Y} \triangleq [a, b] \subset \mathbb{R}$. Naturally, the agent operating in this environment defines an interventional distribution P_π summarizing the consequences of its actions.

Fig. 1a shows a graphical representation (for now, without the highlighted bi-directed arrows) of this data-generating process where nodes represent observed variables and directed arrows represent the functional relationships between them. For every time step $t > 1$, the current state S_t "block" all pathways from previous nodes (e.g., S_{t-1}) to the future nodes (e.g., S_{t+1}) [24, Def. 1.2.3]. Applying the d-separation rules leads to the following independence relationships in distribution P_π.

Definition 1 (Markov Property [29])**.** For a joint distribution P_* over a sequence of states $S_1, S_2, \dots,$ actions $X_1, X_2, \dots,$ and rewards $Y_1, Y_2, \dots,$ the Markov property if for every time step $t = 1, 2, \dots,$ $\left(\bar{\boldsymbol{S}}_{1:t-1}, \bar{\boldsymbol{X}}_{1:t-1}, \bar{\boldsymbol{Y}}_{1:t-1} \perp\!\!\!\perp \bar{\boldsymbol{X}}_{t:\infty}, \bar{\boldsymbol{S}}_{t+1:\infty}, \bar{\boldsymbol{Y}}_{t:\infty} \mid S_t\right)$ holds with regard to distribution P_*.

It follows from Def. 1 that for any horizon T, the distribution generated by a policy π factories as[1]

$$P_\pi\left(\bar{\boldsymbol{x}}_{1:T}, \bar{\boldsymbol{s}}_{1:T}, \bar{\boldsymbol{y}}_{1:T}\right) = P(s_1) \prod_{t=1}^{T} \pi(x_t \mid s_t) \mathcal{T}(s_t, x_t, s_{t+1}) \mathcal{R}(s_t, x_t, y_t) \tag{1}$$

where the transition distribution $\mathcal{T}$ and the reward distribution $\mathcal{R}$ are interventional queries given by

$$\mathcal{T}(s_t, x_t, s_{t+1}) = P_{x_t}\left(s_{t+1} \mid s_t\right) = \int_{\boldsymbol{u}_t} \mathbb{1}_{s_{t+1} = f_S(s_t, x_t, \boldsymbol{u}_t)} P(\boldsymbol{u}_t) \tag{2}$$

$$\mathcal{R}(s_t, x_t, y_t) = P_{x_t}\left(y_t \mid s_t\right) = \int_{\boldsymbol{u}_t} \mathbb{1}_{y = f_Y(s_t, x_t, \boldsymbol{u}_t)} P(\boldsymbol{u}_t) \tag{3}$$

For convenience, we also write the reward function $\mathcal{R}(s, x)$ as the expected value $\sum_y y \mathcal{R}(s, x, y)$. Fix a discounted factor $\gamma \in [0, 1]$ (e.g., the interest rate). A common objective for an agent is to optimize its cumulative return $R_t = \sum_{i=0}^{\infty} \gamma^i Y_{t+i}$. In analysis, we often evaluate the state value function $V_\pi(s)$, which is the expected return given the agent's starting state $S_t = s$. That is, $V_\pi(s) = \mathbb{E}_\pi\left[R_t \mid S_t = s\right]$. A similar state-action value function $Q_\pi(s, x)$ is defined as the expected return starting from state s, taking action x and thereafter following policy π, i.e., $Q_\pi(s, x) = \mathbb{E}_{X_t \leftarrow x, \pi}\left[R_t \mid S_t = s\right]$. One could recursively evaluate the state value function of any state s using the *Bellman Equation* [4]:

$$V_\pi(s) = \sum_x \pi(x \mid s) \left(\mathcal{R}(s, x) + \gamma \sum_{s'} \mathcal{T}(s, x, s') V_\pi(s')\right) \tag{4}$$

Similarly, an analogous equation for the state-action value function is

$$Q_\pi(s, x) = \mathcal{R}(s, x) + \gamma \sum_{s'} \mathcal{T}(s, x, s') V_\pi(s') \tag{5}$$

[1] The decomposition holds since state S_t blocks all backdoor path from action X_t to nodes S_{t+1} and Y_t, i.e., path starting with arrow $X_t \leftarrow S_t$. It follows from Rule 2 of do-calculus [24, Theorem 3.4.1] that $P_\pi\left(y_t \mid s_t, x_t\right) = P_{x_t}\left(s_{t+1} \mid s_t\right)$ and $P_\pi\left(y_t \mid s_t, x_t\right) = P_{x_t}\left(y_t \mid s_t\right)$.

Off-Policy Evaluation Despite the effectiveness of planning algorithms, they require detailed parametrization of the transition distribution $\mathcal{T}$ and the reward function $\mathcal{R}$, which are not accessible in many real-world applications. This means that a learning process must take place. A common approach is off-policy learning, where the agent has access to observed trajectories generated by a *behavioral policy* f_X, different from the target policy π, operating in the same environment. More specifically, for every time step t, the behavioral policy selects an action $X_t \leftarrow f_X(s_t, \boldsymbol{u}_t)$ based on the current state $S_t = s_t$ and latent noise $\boldsymbol{U}_t = \boldsymbol{u}_t$. Fig. 1a shows the graphical representation of the data-generating process of the behavior policy; the added bi-directed arrows, e.g., $X_t \longleftrightarrow Y_t$, indicate the presence of an unobserved confounder $U \in \boldsymbol{U}_t$ affecting both the action X_t and outcome Y_t. We summarize observed trajectories of the behavior policy using the observational distribution P.

Off-policy evaluation attempts to estimate the effects of a candidate policy $\pi(x|s)$ from the observational data generated by the behavior policy f_X. Standard off-policy methods focus on the identifiable setting where the target transition distribution $\mathcal{T}$ and reward function $\mathcal{R}$ remain consistent in both the interventional P_π and observational distribution P. Formally,

Definition 2 (Causal Consistency)**.** For an interventional distribution P_π and an observational distribution P satisfying the Markov property (Def. 1), the Causal Consistency holds with regard to P_π and P if the following statement holds, for every time step $t = 1, 2, \dots,$

$$P_{x_t}\left(s_{t+1} \mid s_t\right) = P\left(s_{t+1} \mid s_t, x_t\right), \quad \text{and} \quad P_{x_t}\left(y_t \mid s_t\right) = P\left(y_t \mid s_t, x_t\right) \tag{6}$$

When Def. 2 holds, the learner could recover the parametrization of the transition distribution $\mathcal{T}$ and reward function $\mathcal{R}$ from the observational data, following the identification formula in Eq. (6). Several off-policy algorithms have been proposed to estimate the effect of candidate policies from finite observations under causal consistency in this identifiable setting [38, 37, 35, 12, 28, 21].

There exist graphical criteria in the literature [25, 31, 26] to evaluate whether causal consistency (Def. 2) holds from causal knowledge of the environment, including the celebrated *backdoor* criterion [24, Def. 3.3.1]. However, in many practical applications, causal consistency could be fragile and does not necessarily hold due to some violations in the generative process. These include: (1) there exists an unobserved confounder affecting the action X_t and subsequent outcomes Y_t, S_{t+1} simultaneously (blue, dashed arrows in Fig. 1a); (2) there is no overlap in the support between the target and behavior policies, i.e., the propensity score $P(x_t \mid s_t) = 0$ for some state-action pair s_t, x_t. When either of these violations occurs, applying standard off-policy methods may fail to recover the expected return of the target policy, leading to estimation bias. The following example illustrates such challenges.

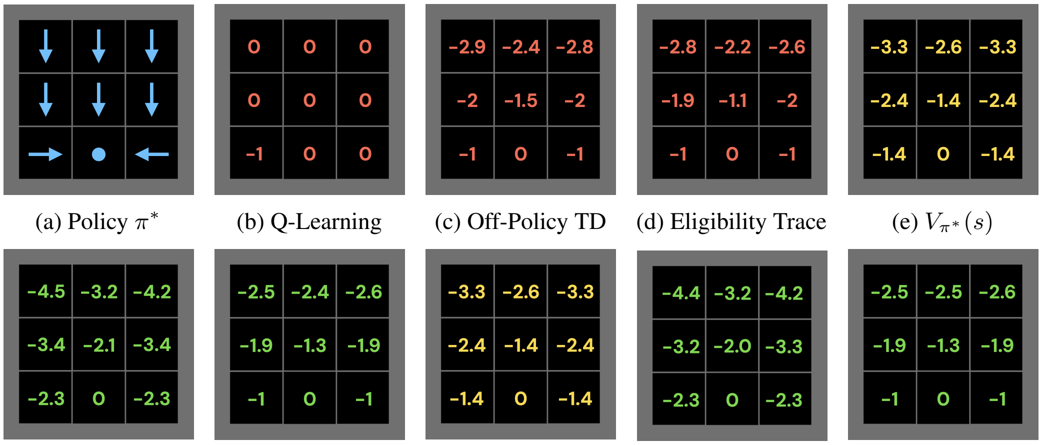
Example 1 (Windy Gridworld)**.** Consider a Windy Gridworld described in Fig. 1b, where the red dot represents the agent and the green square represents the goal state. The agent can take five actions X_t - `up`, `down`, `right`, `left`, and `stay-put`. However, the agent's movement is affected by the wind; the direction of the wind U_t includes - `north`, `south`, `west`, `east`, and `no-wind`. For every time step, the agent receives a constant reward $Y_t \leftarrow -1$. The next state of the agent is shifted by both its action and the wind direction through the mechanism $S_{t+1} \leftarrow S_t + X_t + U_t$.

Our goal is to evaluate the expected return of a target policy π^* described in Fig. 2a, which consistently moves towards the goal state regardless of the wind direction. As an input, we have access to observed trajectories generated by a behavior policy $X_t \leftarrow f_X(S_t, U_t)$, which could sense the wind and select an action accordingly. For example, when the agent is located in the top-left corner ($S_t = (0, 0)$) and the wind is blowing `south` ($U_t = (0, 1)$), the behavior policy will decide to move `right` ($X_t = (1, 0)$) so that the agent could get close to the center ($S_{t+1} = (1, 1)$).

Figs. 2b to 2d shows the value function estimation obtained by standard off-policy methods, including Q-Learning, one-step Temporal Difference (`TD`), and Eligibility Traces (`TD(`λ`)`). We also include in Fig. 2e the ground truth value function computed from the underlying model parameters. The simulation reveals that standard off-policy evaluation deviates from the ground truth return. In this observational data, the wind direction U_t is thus an unobserved confounder affecting both the action X_t and next state S_{t+1}, violating causal consistency. See Appendix B for additional discussions.

2.1 Partial Causal Identification in MDPs

For the remainder of this section, we will introduce partial identification methods for off-policy evaluation that is robust to the unobserved confounding and no-overlap. For every time step $t = 1, 2, \dots,$ let the reward Y_t be bounded in a real interval $[a, b]$. By applying a similar bounding strategy



(a) Policy π^* (b) Q-Learning (c) Off-Policy TD (d) Eligibility Trace (e) $V_{\pi^*}(s)$

(f) $\underline{V_{\pi^*}}(s)$ (g) $\overline{V_{\pi^*}}(s)$ (h) $Q_{\pi^*}(s, x^*)$ (i) $\underline{Q_{\pi^*}}(s, x^*)$ (j) $\overline{Q_{\pi^*}}(s, x^*)$

Figure 2: (a) The target policy π^* selecting an action based on the agent's location. (b - d) Value function estimation was obtained by standard off-policy methods. (e - g) The ground-truth state value function computed from the model parametrization and its lower and upper bounds estimated using the extended Bellman equation in Thm. 1. (h - j) The ground-truth state-action value function computed from the model parametrization for actions $x^* \leftarrow \pi^*(s)$ selected by the target policy and its lower and upper bounds computed from the extended Bellman equation in Thm. 2

in [18, 39, 13], we derive the following bounds over the transition distribution $\mathcal{T}$ and reward function $\mathcal{R}$, for every realization $(s, x, s') \in \mathcal{S} \times \mathcal{X} \times \mathcal{S}$,

$$\mathcal{T}(s, x, s') \in \left[\widetilde{\mathcal{T}}(s, x, s')\, P(x \mid s), \widetilde{\mathcal{T}}(s, x, s')\, P(x \mid s) + P(\neg x \mid s)\right] \tag{7}$$

$$\mathcal{R}(s, x) \in \left[\widetilde{\mathcal{R}}(s, x)\, P(x \mid s) + aP(\neg x \mid s), \widetilde{\mathcal{R}}(s, x)\, P(x \mid s) + bP(\neg x \mid s)\right] \tag{8}$$

Among the above quantities, $P(x \mid s)$ stands for the propensity score $P(X_t = x \mid S_t = s)$ and $P(\neg x \mid s) = 1 - P(x \mid s)$; $\widetilde{\mathcal{T}}$ and $\widetilde{\mathcal{R}}$ are the nominal transition distribution and reward function computed from the observational distribution as follows:

$$\widetilde{\mathcal{T}}(s, x, s') = P(S_{t+1} = s' \mid S_t = s, X_t = x), \qquad \widetilde{\mathcal{R}}(s, x) = \mathbb{E}[Y_t \mid S_t = s, X_t = x] \tag{9}$$

In order to bound the value function $V_\pi(s)$ at state s induced by a candidate policy π, one could minimize/maximize the optimization program using the Bellman's equation in Eq. (4) as the objective function, subject to constraints in Eqs. (7) and (8). Interestingly, this optimization problem is equivalent to a linear program; solving it leads to the following *extended Bellman equation*.

Theorem 1 (Causal Bellman Equation)**.** *For an MDP environment M with reward $Y_t \in [a, b] \subseteq \mathbb{R}$, for any policy $\pi(x \mid s)$, its state value function $V_\pi(s) \in \left[\underline{V_\pi}(s), \overline{V_\pi}(s)\right]$ for every state $s \in \mathcal{S}$, where bounds $\underline{V_\pi}, \overline{V_\pi}$ are solutions given by the following dynamic programs,*

$$\left\langle \underline{V_\pi}(s), \overline{V_\pi}(s) \right\rangle = \sum_x P(x \mid s) \left(\pi(x \mid s) \left(\widetilde{\mathcal{R}}(s, x) + \gamma \sum_{s', x'} \widetilde{\mathcal{T}}(s, x, s') \left\langle \underline{V_\pi}(s'), \overline{V_\pi}(s') \right\rangle \right) \right. \tag{10}$$

$$\left. + \pi(\neg x \mid s) \left(\langle a, b \rangle + \gamma \left\langle \min_{s'} \underline{V_\pi}(s'), \max_{s'} \overline{V_\pi}(s') \right\rangle \right) \right) \tag{11}$$

Thm. 1 can be seen as an extension of the Bellman equation using the confounded observational distribution with no-overlap. For instance, in the lower bound $\underline{V_\pi}(s)$, Eq. (10) follows the standard iterative step in Bellman equation in Eq. (4), measuring the expected return when the target policy's action coincides with the observed action selected by the behavior policy; Eq. (11) could be thought as a regularizing term measuring the uncertainty due to unobserved confounding. Finally, both terms are weighted by the nominal propensity score $P(x \mid s) = P(X_t = x \mid S_t = s)$. The same derivation also applies to the upper bound $\overline{V_\pi}(s)$. An analogous extended Bellman equation bounding the state-action value function from the observational distribution can also be derived as follows.

Theorem 2 (Causal Bellman Equation)**.** *For an MDP environment M with reward signals $Y_t \in [a,b] \subseteq \mathbb{R}$, for any policy $\pi(x \mid s)$, its state-action value function $Q_\pi \in \left[\underline{Q_\pi}(s,x), \overline{Q_\pi}(s,x)\right]$ for any state-action pair $(s,x) \in \mathcal{S} \times \mathcal{X}$, where bounds $\underline{Q_\pi}, \overline{Q_\pi}$ are given by as follows,*

$$\left\langle \underline{Q_\pi}(s,x), \overline{Q_\pi}(s,x) \right\rangle = P(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + \gamma \sum_{s',x'} \widetilde{\mathcal{T}}(s,x,s') \left\langle \underline{V_\pi}(s'), \overline{V_\pi}(s') \right\rangle \right) \tag{12}$$

$$+ P(\neg x \mid s)\left(\langle a,b \rangle + \gamma \left\langle \min_{s'} \underline{V_\pi}(s'), \max_{s'} \overline{V_\pi}(s') \right\rangle \right) \tag{13}$$

Among the bounds in Thm. 2, Eq. (12) is the standard iterative step of the Bellman equation in Eq. (5), weighted by the score $P(x \mid s)$. It estimates the expected return of performing action $\text{do}(x)$ at state s when such action matches the one selected by the behavior policy. Eq. (13) is a regularized term accounting for uncertainties when the intervention $\text{do}(x)$ is not observed in the offline data. Since Thms. 1 and 2 are closed-form solutions of optimization programs and the observational constraints in Eqs. (7) and (8) are tight, the extended Bellman's equation bounds are sharp from offline data and Markov property. This means they cannot be improved without additional assumptions.

Example 2 (Windy Gridworld Continued)**.** Consider again the Windy Gridworld described in Example 1. We compute the lower and upper bounds over the state value function following the extended Bellman equation in Thm. 1, and provide them in Figs. 2f to 2g. We also include in Fig. 2h the ground truth state-action value function for the action $x^* \leftarrow \pi^*(s)$ selected by the target policy. The corresponding lower and upper bounds are shown in Figs. 2i to 2j, following the algorithmic procedure described in Thm. 2. The analysis reveals that the derived bounds are consistent with the ground truth value functions, corroborating the sufficiency of our proposed approach.

3 Confounding Robust Eligibility Traces

The extended Bellman equations described so far require one to have precise estimations for the full models of the nominal transition distribution $\mathcal{T}_{\text{obs}}$, reward function $\mathcal{R}_{\text{obs}}$, and the propensity score $P(x \mid s)$. This section will introduce novel model-free algorithms, using eligibility traces [33], to bound value functions from finite observational samples.

We consider the episodic framework, where the agent interacts with the environment for repeated episodes $n = 1, 2, 3, \dots$; each episode contains a finite number of time steps $t = 1, 2, \dots, T_n$. At each episode, the environment starts at state s_1 following the initial distribution $P(S_1)$. At each time step t, taking the observed state s_t of the environment as input, the behavior policy selects an action x_t. In response to intervention $\text{do}(x_t)$, the environment produces a subsequent reward y_t and moves to the next observed state s_{t+1}. If the next state s_{t+1} is *terminal*, the episode terminates at time step $T_n = t + 1$; the learner receives observational data $\{\bar{\boldsymbol{x}}_{1:T_n-1}, \bar{\boldsymbol{s}}_{1:T_n}, \bar{\boldsymbol{y}}_{1:T_n-1}\}$.

3.1 Causal Temporal Difference

We first introduce a novel augmentation procedure on the celebrated temporal difference (`TD`, [33, 28]) that allows one to estimate the bounds over state value functions, which we call the *causal temporal difference* (`C-TD`). Fig. 3 shows the backup diagram illustrating the idea of our proposed algorithm. Similar to the standard off-policy `TD`, our algorithm will update the estimation of state value functions $\underline{V_\pi}, \overline{V_\pi}$ using the sampled trajectories of transitions in the observational data. It could use a finite number of n-step trajectories or the entire trajectory. Different from the standard off-policy `TD`, our proposed algorithm does not weight each step of the transition using importance sampling (or equivalently, inverse propensity weighting) since the true behavior policy f_X (propensity score) is not recoverable from the observational data. Instead, `C-TD` weights each transition using the target policy π and adjusts for the misalignment between the target and behavior policies using an overestimation/underestimation of value function at state s^*. Such s^* is set as the best-case state associated with the highest value in our current estimation when computing upper bounds and the worst-case state estimate for lower bounds.

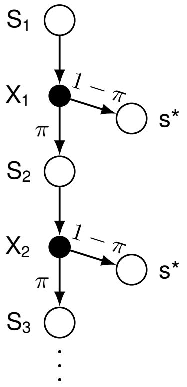


Figure 3: Backup diagram for `C-TD(λ)`.

Algorithm 1 Causal Temporal Difference (`C-TD(λ)`)

```
Require: Observational data $\mathcal{D}$ and a candidate policy $\pi(x \mid s)$.
Update the eligibility traces for all state $s$,
    $e_t(s) = \begin{cases} \gamma\lambda\pi(x_{t-1} \mid s_{t-1})e_{t-1}(s) & \text{if } s \neq s_t \\ \gamma\lambda\pi(x_{t-1} \mid s_{t-1})e_{t-1}(s) + 1 & \text{if } s = s_t \end{cases} \quad (15)$
    where $\lambda \in [0, 1]$ is an eligibility trace decay factor.
Compute the temporal difference error
    $\delta_t = \pi(x_t \mid s_t)\left(y_t + \gamma V_t(s_{t+1})\right) + \pi(\neg x_t \mid s_t)\left(w + \gamma V_t(s^*)\right) - V_t(s_t) \quad (16)$
Update the value function $V_{t+1}(s) \leftarrow V_t(s) + \alpha e_t(s)\delta_t$ for all state $s$.
```

Algorithm 2 Causal Tree-Backup (`C-TB(λ)`)

```
Require: Observational data $\mathcal{D}$ and a candidate policy $\pi(x|s)$.
Update the eligibility traces for all state-action pairs $s, x$,
    $e_t(s, x) = \begin{cases} \gamma\lambda\pi(x_t \mid s_t)\mathbb{1}_{x_{t-1}=x}e_{t-1}(s, x) & \text{if } s \neq s_t \\ \gamma\lambda\pi(x_t \mid s_t)\mathbb{1}_{x_{t-1}=x}e_{t-1}(s, x) + 1 & \text{if } s = s_t \end{cases} \quad (17)$
    where $\lambda \in [0, 1]$ is an eligibility trace decay factor.
Compute the temporal difference error for every action $x$
    $\delta_t(x) = \begin{cases} y_t + \gamma \sum_{x'} \pi(x \mid s_{t+1})Q_t(s_{t+1}, x') - Q_t(s_t, x) & \text{if } x = x_t \\ w + \gamma \sum_{x'} \pi(x' \mid s^*)Q_t(s^*, x') - Q_t(s_t, x) & \text{if } x \neq x_t \end{cases} \quad (18)$
Update the action-value function $Q_{t+1}(s, x) \leftarrow Q_t(s, x) + \alpha e_t(s, x)\delta_t(x)$ for all $s, x$.
```

To formally introduce the estimation algorithm, we first introduce some necessary notations. Let $\boldsymbol{N}(s)$ denote the set of indices of episodes containing a state $s \in \mathcal{S}$, and let $\boldsymbol{t}_n(s)$ be the collection of time steps in the n-th episode such that for every $t \in \boldsymbol{t}_n(s)$, $s_t = s$. For any time step t, let $\pi_t = \pi(x_t \mid s_t)$ and $\neg\pi_t = 1 - \pi(x_t \mid s_t)$. We iteratively define the estimator for bounds over the state value function $V_\pi(s)$ as follows, for any state $s \in \mathcal{S}$,

$$\widehat{V_\pi}(s) = \frac{1}{N} \sum_{n \in \boldsymbol{N}(s)} \sum_{t \in \boldsymbol{t}(s)} \sum_{k=0}^{T_n - t} \gamma^k \Big(\pi_{t+k} y_{t+k} + \neg\pi_{t+k}\big(w + \gamma V(s^*)\big)\Big) \prod_{i=t}^{t+k-1} \pi_i, \tag{14}$$

Among the above equation, N represents the total number of occurrences for the even $s_t = s$ in the observational data. we set parameters $w = a$ and $V(s^*) = \min_s V(s)$ when estimating the lower bound $\underline{V_\pi}(s)$; parameters $w = b$ and $V(s^*) = \max_s V(s)$ for the upper bound $\overline{V_\pi}(s)$.

An eligibility-trace version of our proposed estimation strategy is described Alg. 1. The algorithm keeps track of eligibility traces for every state in a similar manner to standard off-policy temporal difference algorithms. The main difference is that here the eligibility trace is multiplied by the target policy $\pi(x_{t-1} \mid s_{t-1})$ and a decay-rate λ, not including the nominal propensity score $P(x_{t-1} \mid s_{t-1})$. When computing the temporal difference error, the algorithm adjusts for the misalignment between the target and behavior policies by adding a regularized term $w + \gamma V_t(s^*)$, weighted by the probability $1 - \pi(x_t \mid s_t)$. We describe in Alg. 1 a version of `C-TD(λ)` using *online update*. This means that the bounds estimate over value functions are updated at every time step. The *offline* version of the algorithm will use the same temporal difference error and eligibility traces. However, the update only occurs at the end of each episode; the increments and decrements are accumulated on the side, and the value function estimates do not change during the episode.

Theorem 3. *For any behavior policy, for any choice of $\lambda \in [0, 1]$ that does not depend on the actions chosen at each state, let parameters w and s^* be defined as follows: (1) Lower Bound $\underline{V_\pi}$: $w = a$ and $s^* = \arg\min_s V_t(s)$; (2) Upper Bound $\overline{V_\pi}$: $w = b$ and $s^* = \arg\max_s V_t(s)$. Then, Alg. 1 with offline updating converges with probability 1 to lower bound $\underline{V_\pi}$ and upper bound $\overline{V_\pi}$, respectively, under the usual step-size conditions on α.*

The proof of Thm. 3 first shows a contraction property for estimates $\widehat{V}_\pi$, and then follows an application of the general convergence theorem introduced in [11].

3.2 Causal Tree Backup

The algorithm described so far focuses on the estimation of the state value functions. We next introduce a novel algorithm to bound the state-action value function Q_π from finite observations.

Our algorithm is based on an augmentation on the standard tree backup (`TB` [28]), which we call the *causal tree backup* (`C-TB(`λ`)`). The main idea of this new algorithm is illustrated in the backup diagram of Fig. 4. Similar to the standard tree backup, our algorithm updates the value estimates for the action selected by the behavior policy at each time step based on the subsequent reward and the current estimation for the value of the next state. The algorithm then forms a new estimate for the target value function, using the old value estimates for the actions not observed in the observational data and the new estimated value for t-he action taken by the behavior policy. On the other hand, the main differences include the following. (1) Eligibility traces will not only be weighted by the target policy $\pi(x_t \mid s_t)$ using the observed trajectories, but also an indicator function $\mathbb{1}_{x_{t-1}=x}$ returning 1 if the previous action x_{t-1} coincides with the target action x. (2) When the behavior policy takes the same action $x_t = x$ as the target action, the update follows standard `TB` and uses the next sampled state s_t; when the sampled action $x_t \neq x$ differs from the target, our algorithm updates, instead, using the value function associated with the next worst-case or best-case state s^*, corresponding to the estimation of the lower bound and upper bound respectively. The n-step causal tree-backup estimator is defined as

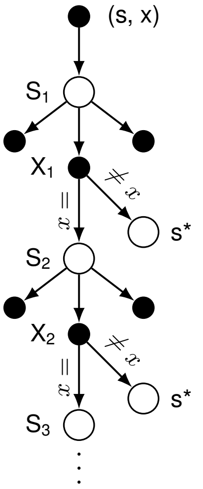


Figure 4: Backup diagram for `C-TB(`λ`)`.

$$\widehat{Q_\pi}(s,x) = \frac{1}{N}\sum_{n\in \boldsymbol{N}(s)}\sum_{t\in \boldsymbol{t}(s)} \gamma^n Q(s_{t+n}, x_{t+n}) \prod_{i=t}^{t+n-1} \pi_{i+1}\mathbb{1}_{x_i=x} + \sum_{k=t}^{t+n}\gamma^{k-t+1}\prod_{i=t}^{t+k-1}\pi_{i+1}\mathbb{1}_{x_i=x}$$

$$\cdot \left(\mathbb{1}_{x_k=x}\left(y_k + \sum_{x'\neq x}\pi(x' \mid s_{k+1})Q(s_{k+1},x')\right) + \mathbb{1}_{x_k\neq x}\left(w + \sum_{x'}\pi(x'\mid s^*)Q(s^*,x')\right)\right) \quad (19)$$

The above tree backup estimator also has a simple incremental implementation using eligibility traces. An online version of this implementation is shown in Fig. 4.

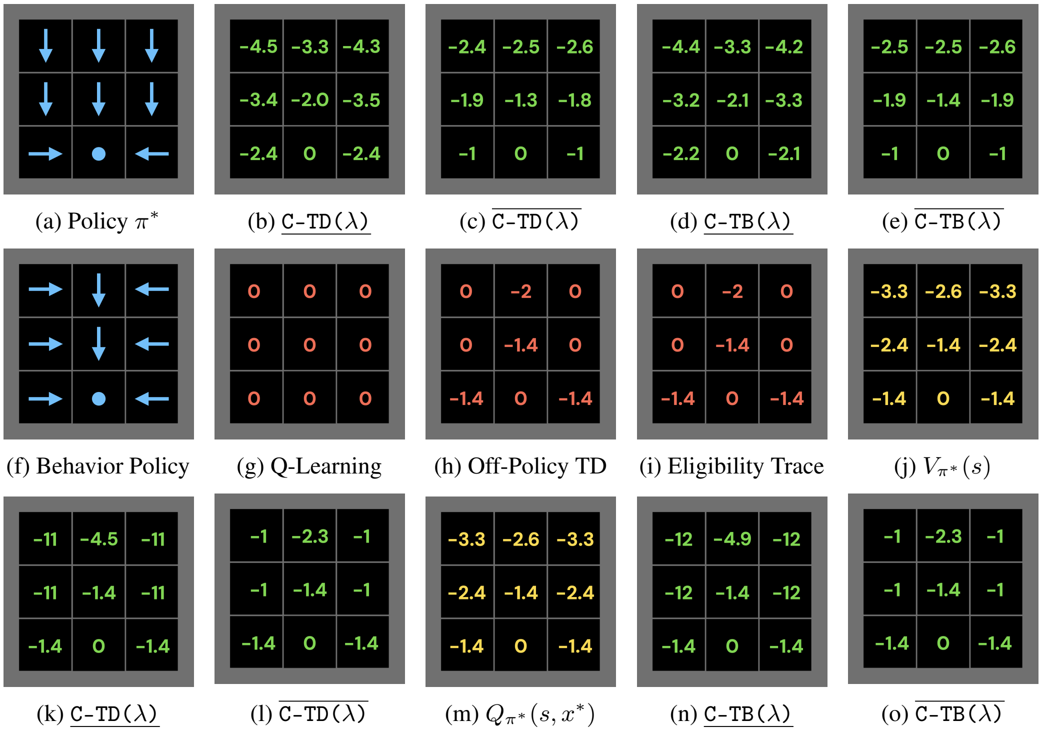
Theorem 4. *For any behavior policy, for any choice of* $\lambda \in [0,1]$ *that does not depend on the actions chosen at each state, let parameters* w *and* s^* *be defined as follows: (1) Lower Bound* $\underline{Q_\pi}$*:* $w = a$ *and* $s^* = \arg\min_s \sum_{x'} \pi(x' \mid s)Q_t(s,x')$*; (2) Upper Bound* $\overline{Q_\pi}$*:* $w = b$ *and* $s^* = \arg\max_s \sum_{x'} \pi(x' \mid s)Q_t(s,x')$*. Then, Alg. 2 with offline updating converges with probability* 1 *to lower bound* $\underline{Q_\pi}$ *and upper bound* $\overline{Q_\pi}$*, respectively, under the usual step-size conditions on* α*.*

The proof of the above theorem relies on a contraction property on the estimates $\widehat{Q}_\pi$ and follows from the general convergence theorem in [11].

4 Experiments

We demonstrate our algorithms using different behavior policies in the Windy Gridworld described in Example 1. Overall, we found that simulation results support our findings, and the proposed algorithms consistently obtain informative bounds over value functions. Experiment 1 evaluates the performance of our bounding strategy in the presence of unobserved confounding. Experiment 2 uses data collected from a deterministic sub-optimal policy, violating the overlap. All experiments use 5×10^4 offline observational samples, meaning that error bars are not significant, hence, not explicitly shown; the decay factor $\lambda = 0.5$. See Appendix B for more details on the experimental setup.

Experiment 1. Consider again the learning setting described in Example 1 where the offline data is contaminated with unobserved confounding bias, and the behavior policy selects actions based on the agent's state and the latent wind direction. We apply `C-TD(`λ`)` to derive bounds over the



(a) Policy π^* (b) $\underline{\texttt{C-TD}(\lambda)}$ (c) $\overline{\texttt{C-TD}(\lambda)}$ (d) $\underline{\texttt{C-TB}(\lambda)}$ (e) $\overline{\texttt{C-TB}(\lambda)}$

(f) Behavior Policy (g) Q-Learning (h) Off-Policy TD (i) Eligibility Trace (j) $V_{\pi^*}(s)$

(k) $\underline{\texttt{C-TD}(\lambda)}$ (l) $\overline{\texttt{C-TD}(\lambda)}$ (m) $Q_{\pi^*}(s, x^*)$ (n) $\underline{\texttt{C-TB}(\lambda)}$ (o) $\overline{\texttt{C-TB}(\lambda)}$

Figure 5: Simulation results comparing causally enhanced off-policy algorithms using eligibility traces ($\texttt{C-TD}(\lambda)$ and $\texttt{C-TB}(\lambda)$) with standard off-policy methods. The offline data are collected from (a - e) a confounded behavior policy affected by the unobserved confounder; and (f - o) a deterministic behavior policy following sub-optimal actions.

state value function $V_{\pi^*}(s)$ and provide them in Figs. 5b and 5c. We also compute the bounds over the state-action value function $Q_{\pi^*}(s, x^*)$ for actions $x^* \leftarrow \pi^*(s)$ using $\texttt{C-TB}(\lambda)$; the simulation results are shown in Figs. 5d and 5e. The analysis reveals that our algorithm consistently recovers the closed-form bounds containing the ground-truth value functions, as previously shown in Fig. 2.

Experiment 2. For the Windy Gridworld environment described in Example 1, suppose the data is now collected by a deterministic behavior policy that always first moves towards the center and then moves down toward the goal; its parametrization is provided in Fig. 5f. This means that the overlap does not hold when the agent is located on either side of the top half of the board. We apply standard off-policy algorithms to evaluate the effect of the target policy π^* of Fig. 5a and provide their evaluations in Figs. 5g to 5i. The propensity score is truncated using a small positive real $0 < \epsilon < 1$ if $P(x \mid s) = 0$. We also compute bounds over the target value functions using our proposed algorithms, $\texttt{C-TD}(\lambda)$ and $\texttt{C-TB}(\lambda)$, and provide their evaluations in Figs. 5k to 5l and Figs. 5n to 5o respectively. By comparing with the ground-truth values in Figs. 5j and 5m, we found that $\texttt{C-TD}(\lambda)$ and $\texttt{C-TB}(\lambda)$ can consistently obtain informative bounds; similar to the previous experiment, standard off-policy methods are not robust against no-overlap and deviate significantly from the target effects.

5 Conclusion

This paper investigates off-policy evaluation in Markov Decision Processes from offline data collected by a different behavior policy, where unobserved confounding bias and no-overlap cannot be ruled out *a priori*. This leads to violations of causal consistency (Def. 2), which could pose significant challenges to standard off-policy algorithms. We first extend the celebrated Bellman's equation to derive informative bounds over values functions from the observational data, which are robust against bias due to the presence of unobserved confounding and no-overlap. Based on these extended equations, we propose two novel model-free off-policy algorithms using eligibility traces – one based on the standard temporal difference ($\texttt{C-TD}(\lambda)$), and the other based on the tree-backup ($\texttt{C-TB}(\lambda)$). These algorithms permit us to consistently bound value functions from finite observations.

References

[1] P. Bajari, C. L. Benkard, and J. Levin. Estimating dynamic models of imperfect competition. *Econometrica*, 75(5):1331–1370, 2007.

[2] A. Balke and J. Pearl. Bounds on treatment effects from studies with imperfect compliance. *Journal of the American Statistical Association*, 92(439):1172–1176, September 1997.

[3] E. Bareinboim, J. D. Correa, D. Ibeling, and T. Icard. On pearl's hierarchy and the foundations of causal inference. In *Probabilistic and causal inference: the works of judea pearl*, pages 507–556. 2022.

[4] R. Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

[5] S. T. Berry and G. Compiani. An instrumental variable approach to dynamic models. *The Review of Economic Studies*, 90(4):1724–1758, 2023.

[6] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[7] D. Bruns-Smith and A. Zhou. Robust fitted-q-evaluation and iteration under sequentially exogenous unobserved confounders. *arXiv preprint arXiv:2302.00662*, 2023.

[8] F. A. Bugni. Bootstrap inference in partially identified models defined by moment inequalities: Coverage of the identified set. *Econometrica*, 78(2):735–753, 2010.

[9] M. J. Dickstein and E. Morales. What do exporters know? *The Quarterly Journal of Economics*, 133(4):1753–1801, 2018.

[10] G. W. Imbens and D. B. Rubin. Bayesian inference for causal effects in randomized experiments with noncompliance. *The annals of statistics*, pages 305–327, 1997.

[11] T. Jaakkola, S. Singh, and M. Jordan. Reinforcement learning algorithm for partially observable markov decision problems. *Advances in neural information processing systems*, 7, 1994.

[12] N. Jiang and L. Li. Doubly robust off-policy value evaluation for reinforcement learning. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 652–661, New York, New York, USA, 20–22 Jun 2016. PMLR.

[13] S. Joshi, J. Zhang, and E. Bareinboim. Towards safe policy learning under partial identifiability: A causal approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13004–13012, 2024.

[14] N. Kallus and A. Zhou. Confounding-robust policy improvement. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 9289–9299, 2018.

[15] N. Kallus and A. Zhou. Confounding-robust policy evaluation in infinite-horizon reinforcement learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 22293–22304. Curran Associates, Inc., 2020.

[16] C. Kausik, Y. Lu, K. Tan, M. Makar, Y. Wang, and A. Tewari. Offline policy evaluation and optimization under confounding. In *International Conference on Artificial Intelligence and Statistics*, pages 1459–1467. PMLR, 2024.

[17] S. Khan, M. Saveski, and J. Ugander. Off-policy evaluation beyond overlap: partial identification through smoothness. *arXiv preprint arXiv:2305.11812*, 2023.

[18] C. Manski. Nonparametric bounds on treatment effects. *American Economic Review, Papers and Proceedings*, 80:319–323, 1990.

[19] H. R. Moon and F. Schorfheide. Bayesian and frequentist inference in partially identified models. *Econometrica*, 80(2):755–782, 2012.

[20] E. Morales, G. Sheu, and A. Zahler. Extended gravity. *The Review of economic studies*, 86(6):2668–2712, 2019.

[21] R. Munos, T. Stepleton, A. Harutyunyan, and M. Bellemare. Safe and efficient off-policy reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 1054–1062, 2016.

[22] H. Namkoong, R. Keramati, S. Yadlowsky, and E. Brunskill. Off-policy policy evaluation for sequential decisions under unobserved confounding. *Advances in Neural Information Processing Systems*, 33:18819–18831, 2020.

[23] A. Norets and X. Tang. Semiparametric inference in dynamic binary choice models. *Review of Economic Studies*, 81(3):1229–1262, 2014.

[24] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, 2000.

[25] J. Pearl and J. Robins. Probabilistic evaluation of sequential plans from causal models with hidden variables. In P. Besnard and S. Hanks, editors, *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence (UAI 1995)*, pages 444–453. Morgan Kaufmann, San Francisco, 1995.

[26] E. Perković, J. Textor, M. Kalisch, and M. H. Maathuis. A complete generalized adjustment criterion. *arXiv preprint arXiv:1507.01524*, 2015.

[27] D. J. Poirier. Revising beliefs in nonidentified models. *Econometric theory*, 14(4):483–509, 1998.

[28] D. Precup, R. S. Sutton, and S. P. Singh. Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 759–766, 2000.

[29] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[30] J. P. Romano and A. M. Shaikh. Inference for identifiable parameters in partially identified econometric models. *Journal of Statistical Planning and Inference*, 138(9):2786–2807, 2008.

[31] I. Shpitser, T. VanderWeele, and J. Robins. On the validity of covariate adjustment for estimating causal effects. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pages 527–536. AUAI, Corvallis, OR, 2010.

[32] J. Stoye. More on confidence intervals for partially identified parameters. *Econometrica*, 77(4):1299–1315, 2009.

[33] R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3:9–44, 1988.

[34] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

[35] A. Swaminathan and T. Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, pages 814–823, 2015.

[36] D. Todem, J. Fine, and L. Peng. A global sensitivity test for evaluating statistical hypotheses with nonidentifiable models. *Biometrics*, 66(2):558–566, 2010.

[37] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[38] C. J. C. H. Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge England, 1989.

[39] J. Zhang and E. Bareinboim. Near-optimal reinforcement learning in dynamic treatment regimes. In *Advances in Neural Information Processing Systems*, pages 13401–13411, 2019.

[40] J. Zhang, J. Tian, and E. Bareinboim. Partial counterfactual identification from observational and experimental data. In *International Conference on Machine Learning*, pages 26548–26558. PMLR, 2022.

A Proofs

This section provides proof of the main theoretical results provided in the paper.

Theorem 1 (Causal Bellman Equation)**.** *For an MDP environment M with reward $Y_t \in [a, b] \subseteq \mathbb{R}$, for any policy $\pi(x \mid s)$, its state value function $V_\pi(s) \in \left[\underline{V_\pi}(s), \overline{V_\pi}(s)\right]$ for every state $s \in \mathcal{S}$, where bounds $\underline{V_\pi}$, $\overline{V_\pi}$ are solutions given by the following dynamic programs,*

$$\left\langle \underline{V_\pi}(s), \overline{V_\pi}(s) \right\rangle = \sum_x P(x \mid s) \Bigg(\pi(x \mid s) \Bigg(\widetilde{\mathcal{R}}(s, x) + \gamma \sum_{s', x'} \widetilde{\mathcal{T}}(s, x, s') \left\langle \underline{V_\pi}(s'), \overline{V_\pi}(s') \right\rangle \Bigg) \tag{10}$$

$$+\pi(\neg x \mid s) \Bigg(\langle a, b \rangle + \gamma \left\langle \min_{s'} \underline{V_\pi}(s'), \max_{s'} \overline{V_\pi}(s') \right\rangle \Bigg) \Bigg) \tag{11}$$

Proof. Following the Bellman equation [4], the state value function at state $s \in \mathcal{S}$ is given by

$$V_\pi(s) = \sum_x \pi(x \mid s) \left(\mathcal{R}(s, x) + \gamma \sum_{s'} \mathcal{T}(s, x, s') V_\pi(s') \right) \tag{20}$$

Among the above quantities, the reward function $\mathcal{R}$ is bounded from the observational distribution [18] as follows,

$$\widetilde{\mathcal{R}}(s, x)\, P(x \mid s) + aP(\neg x \mid s) \leq \mathcal{R}(s, x) \leq \widetilde{\mathcal{R}}(s, x)\, P(x \mid s) + bP(\neg x \mid s) \tag{21}$$

where $\widetilde{\mathcal{R}}$ is the nominal reward function computed from the observational distribution and is defined in Eq. (9). Replacing the reward function $\mathcal{R}$ in Eq. (20) with the above lower bound gives

$$\begin{aligned} V_\pi(s) \geq \sum_x \pi(x \mid s) \left(\widetilde{\mathcal{R}}(s, x)\, P(x \mid s) + aP(\neg x \mid s) + \gamma \sum_{s'} \mathcal{T}(s, x, s') V_\pi(s') \right) \\ + \sum_x b\pi(x \mid s) P(\neg x \mid s) \end{aligned} \tag{22}$$

Similarly, the transition distribution $\mathcal{T}$ can be bounded from the observational distribution [18],

$$\widetilde{\mathcal{T}}(s, x, s')\, P(x \mid s) \leq \mathcal{T}(s, x, s') \leq \widetilde{\mathcal{T}}(s, x, s')\, P(x \mid s) + P(\neg x \mid s) \tag{23}$$

and $\widetilde{\mathcal{T}}$ is the nominal transition distribution computed from the observational distribution defined in Eq. (9). Minimizing the lower bound in Eq. (22) subject to the above observational constraints in Eq. (23) and $\sum_{s'} \mathcal{T}(s, x, s') = 1$ gives the following lower bound:

$$\begin{aligned} V_\pi(s) \geq \sum_x \pi(x \mid s) P(x \mid s) \left(\widetilde{\mathcal{R}}(s, x) + aP(\neg x \mid s) + \gamma \sum_{s'} \widetilde{\mathcal{T}}(s, x, s') V_\pi(s') \right) \\ + \sum_x \pi(x \mid s) P(\neg x \mid s) \left(b + \min_{s'} V_\pi(s') \right) \end{aligned} \tag{24}$$

The above lower bound is achieved by setting the worst-case transition probability $\mathcal{T}(s, x, s^*) = P(\neg x \mid s)$ for state $s^* = \arg\min_{s'} V_\pi(s')$ and $\mathcal{T}(s, x, s') = \widetilde{\mathcal{T}}(s, x, s') P(x \mid s)$ for all the other state $s' \neq s^*$. Note that the second term of the above inequality could be further written as:

$$\sum_x \pi(x \mid s) P(\neg x \mid s) \left(a + \min_{s'} V_\pi(s') \right) \tag{25}$$

$$= \sum_x \pi(x \mid s) \left(1 - P(x \mid s)\right) \left(a + \min_{s'} V_\pi(s') \right) \tag{26}$$

$$= \sum_x \pi(x \mid s) \left(a + \min_{s'} V_\pi(s') \right) - \sum_x \pi(x \mid s) P(x \mid s) \left(a + \min_{s'} V_\pi(s') \right) \tag{27}$$

$$= \sum_x P(x \mid s) \left(a + \min_{s'} V_\pi(s') \right) - \sum_x \pi(x \mid s) P(x \mid s) \left(a + \min_{s'} V_\pi(s') \right) \tag{28}$$

The last step holds since for any constant real value C, $\sum_x \pi(x \mid s)C = \sum_x P(x \mid s)C$. The above equation can be further written as

$$\sum_x \pi(x \mid s)P(\neg x \mid s)\left(a + \min_{s'} V_\pi(s')\right) = \sum_x \pi(\neg x \mid s)P(x \mid s)\left(a + \min_{s'} V_\pi(s')\right) \tag{29}$$

Replacing the second term in Eq. (24) gives

$$\begin{aligned} V_\pi(s) \geq \sum_x \pi(x \mid s)P(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + bP(\neg x \mid s) + \gamma \sum_{s'} \widetilde{\mathcal{T}}(s,x,s')V_\pi(s')\right) \\ + \sum_x \pi(\neg x \mid s)P(x \mid s)\left(a + \min_{s'} V_\pi(s')\right) \end{aligned} \tag{30}$$

After a few simplifications, we obtain

$$\begin{aligned} V_\pi(s) \geq P(x \mid s)\Bigg(\pi(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + \gamma \sum_{s',x'} \widetilde{\mathcal{T}}(s,x,s')\, V_\pi(s')\right) \\ + \pi(\neg x \mid s)\left(a + \gamma \min_{s'} V_\pi(s')\right)\Bigg) \end{aligned} \tag{31}$$

Finally, minimizing the value function V_π subject to the above inequality gives the lower bound $\underline{V_\pi}$. The upper bound $\overline{V_\pi}$ over the state value function could be similarly derived. □

Theorem 2 (Causal Bellman Equation)**.** *For an MDP environment M with reward signals $Y_t \in [a,b] \subseteq \mathbb{R}$, for any policy $\pi(x \mid s)$, its state-action value function $Q_\pi \in \left[\underline{Q_\pi}(s,x), \overline{Q_\pi}(s,x)\right]$ for any state-action pair $(s,x) \in \mathcal{S} \times \mathcal{X}$, where bounds $\underline{Q_\pi}, \overline{Q_\pi}$ are given by as follows,*

$$\left\langle \underline{Q_\pi}(s,x), \overline{Q_\pi}(s,x)\right\rangle = P(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + \gamma \sum_{s',x'} \widetilde{\mathcal{T}}(s,x,s')\left\langle \underline{V_\pi}(s'), \overline{V_\pi}(s')\right\rangle\right) \tag{12}$$

$$+P(\neg x \mid s)\left(\langle a,b\rangle + \gamma\left\langle \min_{s'} \underline{V_\pi}(s'), \max_{s'} \overline{V_\pi}(s')\right\rangle\right) \tag{13}$$

Proof. Applying Bellman equation [4] allows us to iteratively write the state-action value function for any state-action pair $(s,x) \in \mathcal{S} \times \mathcal{X}$ as

$$Q_\pi(s,x) = \mathcal{R}(s,x) + \gamma \sum_{s'} \mathcal{T}(s,x,s')V_\pi(s') \tag{32}$$

where the reward function $\mathcal{R}$ is bounded from the observational distribution [18] following Eq. (21). Replacing the reward function $\mathcal{R}$ in the above equation with the corresponding lower bound gives

$$Q_\pi(s,x) \geq P(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + \gamma \sum_{s'} \mathcal{T}(s,x,s')V_\pi(s')\right) + aP(\neg x \mid s) \tag{33}$$

Similarly, the transition distribution $\mathcal{T}$ can be bounded from the observational distribution [18] following Eq. (23). Minimizing the lower bound in Eq. (33) subject to the above observational constraints in Eq. (23) and $\sum_{s'} \mathcal{T}(s,x,s') = 1$ gives the following solution:

$$Q_\pi(s,x) \geq P(x \mid s)\left(\widetilde{\mathcal{R}}(s,x) + \gamma \sum_{s'} \widetilde{\mathcal{T}}(s,x,s')V_\pi(s')\right) + P(\neg x \mid s)\left(a + \min_{s'} V_\pi(s')\right) \tag{34}$$

This lower bound is achieved by setting the worst-case transition probability $\mathcal{T}(s,x,s^*) = P\left(\neg x \mid s\right)$ for state $s^* = \arg\min_{s'} V_\pi(s')$ and $\mathcal{T}(s,x,s') = \widetilde{\mathcal{T}}(s,x,s')P\left(x \mid s\right)$ for all the other state $s' \neq s^*$. Finally, notice that $V_\pi(s)$ is a function of $Q_\pi(s,x)$ and is given by $V_\pi(s) = \sum_x \pi(x \mid s)Q_\pi(s,x)$. Minimizing the state-action value function Q_π subject to the above inequality leads to the lower bound $\underline{Q_\pi}$. The upper bound $\overline{Q_\pi}$ could be similarly derived. □

Theorem 3. *For any behavior policy, for any choice of $\lambda \in [0, 1]$ that does not depend on the actions chosen at each state, let parameters w and s^* be defined as follows: (1) Lower Bound $\underline{V_\pi}$: $w = a$ and $s^* = \arg\min_s V_t(s)$; (2) Upper Bound $\overline{V_\pi}$: $w = b$ and $s^* = \arg\max_s V_t(s)$. Then, Alg. 1 with offline updating converges with probability 1 to lower bound $\underline{V_\pi}$ and upper bound $\overline{V_\pi}$, respectively, under the usual step-size conditions on α.*

Proof. We will focus on the convergence of lower bound $\underline{V_\pi}(s)$; the proof for the upper bound $\overline{V_\pi}(s)$ follows analogously. The proof is structured in two stages. First, we consider the truncated lower bound estimates corresponding to Eq. (14), which sums the adjusted rewards obtained from the environment for only n steps, then uses the current estimate of the value function lower bound to approximate the remaining value:

$$\underline{R_t}^{(n)} = \sum_{k=0}^{n-1} \gamma^k \Big(\pi_{t+k} y_{t+k} + \neg\pi_{t+k} \big(b + \gamma \min_{s'} V(s')\big)\Big) \prod_{i=t}^{t+k-1} \pi_i + \gamma^n V(s_{t+n}) \prod_{i=t}^{t+k-1} \pi_i \tag{35}$$

We need to show that $\underline{R_t}^{(n)} - \underline{V_\pi}$ is a contraction mapping in the max norm. If this is true for any n, then by applying the general convergence theorem, the n-step return converges to $\underline{V_\pi}$. Then any convex combination will also converge to $\underline{V_\pi}$. For example, any combination using a λ parameter in the style of eligibility traces will converge to $\underline{V_\pi}$.

The expected value of the adjusted return with regard to the observational distribution for state s can be expressed as follows [2]:

$$\mathbb{E}\left[\underline{R_t}^{(n)} \mid S_t = s\right] \tag{36}$$

$$= \sum_{k=1}^{n} \sum_{\bar{\boldsymbol{s}}_{1:k}, \bar{\boldsymbol{x}}_{1:k}, \bar{\boldsymbol{y}}_{1:k}} P\left(\bar{\boldsymbol{s}}_{1:k}, \bar{\boldsymbol{x}}_{1:k}, \bar{\boldsymbol{y}}_{1:k}\right) \gamma^{k-1} \left(\pi_k y_k + \neg\pi_k \left(b + \min_{s'} V(s')\right)\right) \prod_{i=1}^{k-1} \pi_i \tag{37}$$

$$+ \sum_{\bar{\boldsymbol{s}}_{1:n}, \bar{\boldsymbol{x}}_{1:n}} P\left(\bar{\boldsymbol{s}}_{1:n}, \bar{\boldsymbol{x}}_{1:n}\right) \gamma^n V(s_n) \prod_{i=1}^{n-1} \pi_i \tag{38}$$

$$= \sum_{k=1}^{n} \gamma^{k-1} \sum_{\bar{\boldsymbol{s}}_{1:k}, \bar{\boldsymbol{x}}_{1:k}} \prod_{i=1}^{k-1} \widetilde{T}(s_i, x_i, s_{i+1}) P(x_i \mid s_i) \pi(x_i \mid s_i) \tag{39}$$

$$\cdot \left(\pi(x_k \mid s_k) \widetilde{\mathcal{R}}(s_k, x_k) + \neg\pi(x_k \mid s_k) \left(b + \gamma \min_{s'} V(s')\right)\right) \tag{40}$$

$$+ \gamma^n \sum_{\bar{\boldsymbol{s}}_{1:n}, \bar{\boldsymbol{x}}_{1:n}} \prod_{i=1}^{n-1} \widetilde{T}(s_i, x_i, s_{i+1}) P(x_i \mid s_i) \pi(x_i \mid s_i) V(s_n) \tag{41}$$

By applying the extended Bellman equation for the lower bound $\underline{V_\pi}$ iteratively n times, we obtain:

$$\underline{V_\pi}(s) = \sum_{k=1}^{n} \sum_{\bar{\boldsymbol{s}}_{1:k}, \bar{\boldsymbol{x}}_{1:k}} \gamma^{k-1} \prod_{i=1}^{k-1} \widetilde{T}(s_i, x_i, s_{i+1}) P(x_i \mid s_i) \pi(x_i \mid s_i) \tag{42}$$

$$\cdot \left(\pi(x_k \mid s_k) \widetilde{\mathcal{R}}(s_k, x_k) + \neg\pi(x_k \mid s_k) \left(b + \gamma \min_{s'} \underline{V_\pi}(s')\right)\right) \tag{43}$$

$$+ \gamma^n \sum_{\bar{\boldsymbol{s}}_{1:n}, \bar{\boldsymbol{x}}_{1:n}} \prod_{i=1}^{n-1} \widetilde{T}(s_i, x_i, s_{i+1}) P(x_i \mid s_i) \pi(x_i \mid s_i) \underline{V_\pi}(s_n) \tag{44}$$

Therefore,

$$\max_s \left| \mathbb{E}\left[\underline{R_t}^{(n)} \mid S_t = s\right] - \underline{V_\pi}(s) \right| \leq \gamma \max_s \left| V(s) - \underline{V_\pi}(s) \right| \tag{45}$$

This means that any n-step return is a contraction in the max norm, and therefore, by applying [11, Theorem 1], it converges to $\underline{V_\pi}(s)$.

[2] We abuse notation a bit and ignore the expected value operator $\mathbb{E}[\cdot]$ outside.

In the second stage, we show that by applying the updates of Alg. 1 for n successive steps, we perform the same update by using the n-step adjusted return $\underline{R_t}^{(n)}$. The eligibility trace for state s can be written as, for $t_n \in \boldsymbol{t}(s)$,

$$e_t(s) = \gamma^{t-t_n} \prod_{i=t_n+1}^{t} \pi_i. \tag{46}$$

We have

$$\sum_{k=1}^{n} e_{t+k-1}(s)\delta_{t+k-1}(s) \tag{47}$$

$$= \sum_{k=1}^{n} \gamma^{k-1} \prod_{i=t+1}^{t+k-1} \pi_i \Big(\pi_{t+k} \left(y_{t+k} + \gamma V(s_{t+k})\right) + \pi_{t+k} \Big(b + \gamma \min_{s'} V(s')\Big) \tag{48}$$

$$- V(s_{t+k-1})\Big) \tag{49}$$

$$= \sum_{k=0}^{n-1} \gamma^k \Big(\pi_{t+k} y_{t+k} + \neg\pi_{t+k} \big(b + \gamma \min_{s'} V(s')\big)\Big) \prod_{i=t}^{t+k-1} \pi_i + \gamma^n V(s_{t+n}) \prod_{i=t}^{t+k-1} \pi_i \tag{50}$$

$$- V(s_t) \tag{51}$$

$$= \underline{R_t}^{(n)} - V(s_t) \tag{52}$$

Since C-TD(λ) is equivalent to applying a convex mixture of n-step updates, and each update converges to correct lower bounds $\underline{V_\pi}$ for the state value functions, Alg. 1 converges to correct lower bounds as well. □

Theorem 4. *For any behavior policy, for any choice of $\lambda \in [0, 1]$ that does not depend on the actions chosen at each state, let parameters w and s^* be defined as follows: (1) Lower Bound $\underline{Q_\pi}$: $w = a$ and $s^* = \arg\min_s \sum_{x'} \pi(x' \mid s) Q_t(s, x')$; (2) Upper Bound $\overline{Q_\pi}$: $w = b$ and $s^* = \arg\max_s \sum_{x'} \pi(x' \mid s) Q_t(s, x')$. Then, Alg. 2 with offline updating converges with probability* 1 *to lower bound $\underline{Q_\pi}$ and upper bound $\overline{Q_\pi}$, respectively, under the usual step-size conditions on α.*

Proof. We will focus on the convergence of lower bound $\underline{Q_\pi}(s, x)$; the proof for the upper bound $\overline{Q_\pi}(s, x)$ follows analogously. This proof is structured in two stages. Let Q_n denote the n-step tree backup estimator defined in Eq. (19). First we show that $\mathbb{E}\left[Q_n(s, x)\right] - \underline{Q_\pi}(s, x)$ is a contraction using a proof by induction.

Let Q be the current estimate of the lower bound for the value function. For $n = 1$,

$$\max_{s,x} \left|\mathbb{E}\left[Q_1(s, x)\right] - \underline{Q_\pi}(s, x)\right| \tag{53}$$

$$= \max_{s,x} \Bigg| P(x \mid s) \Big(\widetilde{\mathcal{R}}(s, x) + \gamma \sum_{s', x'} \widetilde{\mathcal{T}}(s, x, s') \sum_{x'} \pi(x' \mid s') Q(s', x')\Big) \tag{54}$$

$$+ P(\neg x \mid s) \Big(b + \gamma \min_{s'} \sum_{x'} \pi(x' \mid s') Q(s', x')\Big) \tag{55}$$

$$- P(x \mid s) \Big(\widetilde{\mathcal{R}}(s, x) + \gamma \sum_{s', x'} \widetilde{\mathcal{T}}(s, x, s') \sum_{x'} \pi(x' \mid s') \underline{Q_\pi}(s', x')\Big) \tag{56}$$

$$- P(\neg x \mid s) \Big(b + \gamma \min_{s'} \sum_{x'} \pi(x' \mid s') \underline{Q_\pi}(s', x')\Big) \Bigg| \tag{57}$$

$$\leq \gamma \max_{s,x} \left|Q(s, x) - \underline{Q_\pi}(s, x)\right| \tag{58}$$

For the induction step, we assume that

$$\max_{s,x} \left|\mathbb{E}\left[Q_n(s, x)\right] - \underline{Q_\pi}(s, x)\right| \leq \gamma \max_{s,x} \left|Q(s, x) - \underline{Q_\pi}(s, x)\right| \tag{59}$$

Next we want to show that the same holds for $Q_{n+1}(s, x)$. We can rewrite $Q_{n+1}(s, x)$ as follows,

$$Q_{n+1}(s,x) = \mathbb{1}_{x_t=x}\left(y_t + \sum_{x'}\left(\mathbb{1}_{x'\neq x}\pi(x' \mid s_{t+1})Q(s_{t+1},x') + \mathbb{1}_{x'=x}Q_n(s_{t+1},x)\right)\right) \tag{60}$$

$$+ \mathbb{1}_{x_t\neq x}\left(w + \sum_{x'}\pi(x' \mid s^*)Q(s^*,x')\right) \tag{61}$$

We must have

$$\max_{s,x}\left|\mathbb{E}\left[Q_{n+1}(s,x)\right] - \underline{Q_{\pi}}(s,x)\right| \tag{62}$$

$$= \max_{s,x}\Bigg|P(x \mid s)\Bigg(\widetilde{\mathcal{R}}\left(s,x\right) + \gamma\sum_{s',x'}\widetilde{\mathcal{T}}\left(s,x,s'\right)\sum_{x'}\pi(x' \mid s') \tag{63}$$

$$\mathbb{1}_{x'\neq x}Q(s',x') + \mathbb{1}_{x'=x}\mathbb{E}\left[Q_n(s',x)\right]\Bigg) \tag{64}$$

$$+ P(\neg x \mid s)\left(b + \gamma\min_{s'}\sum_{x'}\pi(x' \mid s')Q(s',x')\right) \tag{65}$$

$$- P(x \mid s)\left(\widetilde{\mathcal{R}}\left(s,x\right) + \gamma\sum_{s',x'}\widetilde{\mathcal{T}}\left(s,x,s'\right)\sum_{x'}\pi(x' \mid s')\underline{Q_{\pi}}(s',x')\right) \tag{66}$$

$$- P(\neg x \mid s)\left(b + \gamma\min_{s'}\sum_{x'}\pi(x' \mid s')\underline{Q_{\pi}}(s',x')\right)\Bigg| \tag{67}$$

$$\leq \gamma\max_{s,x}\Bigg|P(x \mid s)\gamma\sum_{s',x'}\widetilde{\mathcal{T}}\left(s,x,s'\right)\sum_{x'}\pi(x' \mid s')\mathbb{1}_{x'\neq x}\left(Q(s',x') - \underline{Q_{\pi}}(s',x')\right) \tag{68}$$

$$+ \mathbb{1}_{x'=x}\mathbb{E}\left[\left(Q_n(s',x) - \underline{Q_{\pi}}(s',x')\right)\right] \tag{69}$$

$$+ P(\neg x \mid s)\min_{s'}\sum_{x'}\pi(x' \mid s')\left(Q(s',x') - \underline{Q_{\pi}}(s',x')\right)\Bigg| \tag{70}$$

$$\leq \gamma\max_{s,x}\left|Q(s,x) - \underline{Q_{\pi}}(s,x)\right| \tag{71}$$

By applying [11, Theorem 1], we can conclude that any n-step adjusted return converges to the correct lower bound for the state-action value function. Since all the n-step returns converge to $\underline{Q_{\pi}}$, any convex linear combination of n-step returns also converges to $\underline{Q_{\pi}}$.

For the second part of the proof, we show that `C-TB(`λ`)` with $\lambda = 1$ for n steps is equivalent to using Q_n. The eligibility trace for a state-action pair (s, x) can be rewritten as:

$$e_t(s,x) = \gamma^k\prod_{i=t+1}^{t+k-1}\pi_{i+1}\mathbb{1}_{x_i=x}. \tag{72}$$

By adding and subtracting the weighted action value $\pi_{t+k}\mathbb{1}_{x_{t+k}=x}$ for the action taken on each step from the return, and regrouping, we have

$$Q(s_t,x) + \sum_{k=1}^{n}\gamma^{k-1}\prod_{i=t+1}^{t+k-1}\pi_{i+1}\mathbb{1}_{x_i=x}\Bigg(\mathbb{1}_{x_{t+k}=x}\left(y_{t+k} + \sum_{x'\neq x}\pi(x' \mid s_{t+k+1})Q(s_{t+k+1},x')\right) \tag{73}$$

$$+ \mathbb{1}_{x_{t+k}\neq x}\left(w + \min_{s'}\sum_{x'}\pi(x' \mid s')Q(s',x')\right) - Q(s_{t+k},x)\Bigg) \tag{74}$$

$$= Q(s_t,x) + \sum_{k=1}^{n}e_{t+k}(s_t,x)\delta_{t+k}(x) \tag{75}$$

This concludes the proof. □

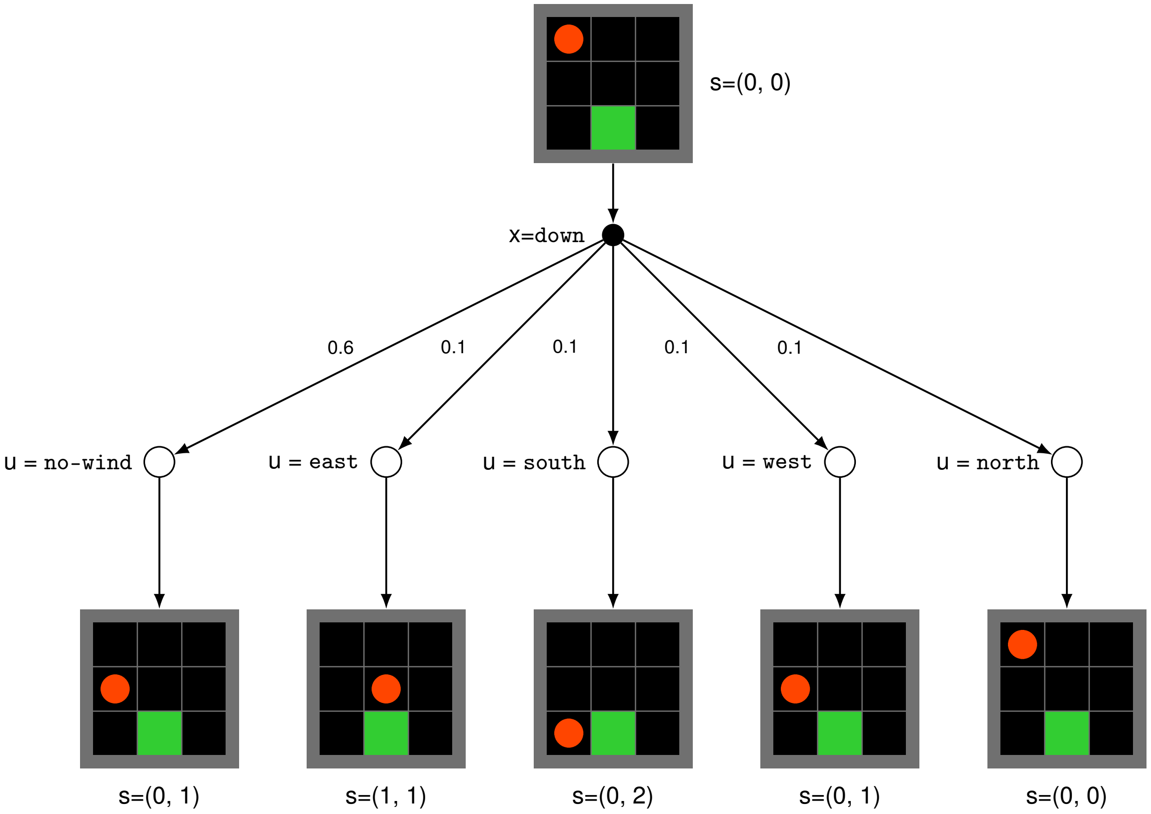
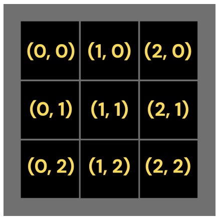


Figure 6: Trajectories sampled from the interventional transition distribution $\mathcal{T}$.

B Experimental Setups

In this section, we provide details on the experimental setups and additional discussion on the simulation environment. All experiments were performed on a 2021 MacBook Pro with 16GB memory, implemented in Python. The simulation environment is built upon the Gymnasium framework [6]. We plan to release the source code with the camera-ready version of the manuscript.



(0, 0)	(1, 0)	(2, 0)
(0, 1)	(1, 1)	(2, 1)
(0, 2)	(1, 2)	(2, 2)

Figure 7: Agent's state in Windy Gridworld environment.

Windy Gridworld Our simulation builds on the Windy Gridworld environment described in Fig. 1b, where the red dot represents the agent and the green square represents the goal state. The agent's location is represented using a vector (i, j) where $i \in \{0, 1, 2\}$ is the column index, and $j \in \{0, 1, 2\}$ is the row index. So the agent's starting state is $(0, 0)$ and the goal state is $(1, 2)$. Fig. 7 shows the detailed state representation for each location in the gridworld.

The agent can take five actions $x \in \mathcal{X}$ - `up`, `down`, `right`, `left`, and `stay-put`, corresponding to vector $(0, -1)$, $(0, 1)$, $(1, 0)$, $(-1, 0)$, and $(0, 0)$ respectively. Meanwhile, the agent's movement is also affected by a wind; the wind direction $u \in \mathcal{U}$ include - `north`, `south`, `east`, `west`, and `no-wind`, corresponding to vector $(0, -1)$, $(0, 1)$, $(1, 0)$, $(-1, 0)$, and $(0, 0)$ respectively. Table 1 summarizes the detailed parametrization for the agent's action and the wind direction. For every time step $t = 1, 2, \ldots,$ the wind U_t can blow in directions `north`, `south`, `east`, `west` with equal probabilities of 10%; otherwise, the weather is nice and there is `no-wind`. That is,

$$\forall i \in \{-1, 1\}, \quad P\left(U_t = (i, 0)\right) = P\left(U_t = (0, i)\right) = 0.1, \quad \text{and} \quad P\left(U_t = (0, 0)\right) = 0.6 \qquad (76)$$

At every time step t, the agent receives a constant reward $Y_t \leftarrow -1$. The next state of the agent is shifted by both its action and the wind direction through the mechanism

$$S_{t+1} \leftarrow \max\left\{\min\left\{S_t + X_t + U_t, (2, 2)\right\}, (0, 0)\right\}. \qquad (77)$$

Action x	up	down	right	left	stay-put
Wind u	north	south	east	west	no-wind
Vector v	$(0,-1)$	$(0,1)$	$(1,0)$	$(-1,0)$	$(0,0)$

Table 1: Vector representations for the agent's action X and the wind direction U.

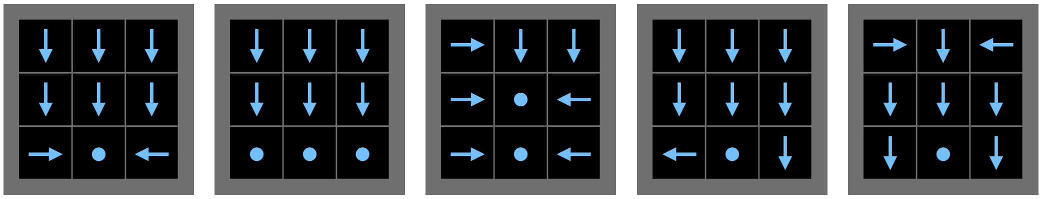
In other words, the agent's next state S_{t+1} is a vector sum of the agent's current location S_t, its action X_t, and the wind direction U_t, truncated by the board's boundary $i = 0, 2$ and $j = 0, 2$. For instance, we show in Fig. 6 the system dynamics for the agent's interactions with the gridworld environment at from the location $s = (0, 0)$, taking the action `down` ($x = (0, 1)$). In this case, when the wind is blowing towards `south` ($u = (0, 1)$), the agent's location will be shifted by both the action x and the windy direction u, and moves to the bottom left corner $s' = (0, 2)$ at the next time step. Since among all wind directions, $u =$ `east` is the only latent state moving the agent to the center $s' = (0, 2)$, we must have the following evaluation for the interventional distribution $P_{X_t}(S_{t+1} \mid S_t)$,

$$\begin{aligned} P_{X_t \leftarrow (0,1)}(S_{t+1} = (0, 2) \mid S_t = (0, 0)) &= P(U_t = (1, 0)) &(78)\\ &= 0.1 &(79)\end{aligned}$$

That is, the agent's transition distribution $\mathcal{T}(s, x, s') = 0.1$ when starting from $s = (0, 1)$, taking action $x = (0, 1)$, and moving to the next state $s' = (0, 2)$.

Confounded Behavior Policy Consider now an off-policy learning task in the windy gridworld, where the agent's goal is to evaluate the expected return of a target policy π^* described in Fig. 2a. Following such a policy π^*, the agent will consistently move towards the goal state $s = (1, 2)$ from its current location, regardless of the wind direction.

The detailed parametrization of the agent's system dynamics in the windy gridworld remains unknown. Instead, its has access to observed trajectories generated by a behavior policy $x \leftarrow f_X(s, u)$ which could sense the wind and select an action accordingly; Fig. 8 provides a detailed description for this behavior policy. For example, when the agent is located in the top-left corner ($s = (0, 0)$) and the wind is blowing `south` ($s = (0, 1)$), the behavior policy $x \leftarrow f_X(s, u)$ will decide to move `right` ($x = (1, 0)$) so that the agent could get to the center ($s' = (1, 1)$).



(a) `no wind` (b) `east` (c) `south` (d) `west` (e) `north`

Figure 8: A confounded behavior policy f_X selecting values based on the agent's location S and the latent wind direction U.

Consequently, the wind direction U_t becomes an unobserved confounder in the generative process for the offline observational data, affecting the allocated action X_t and the next state S_{t+1} simultaneously. The presence of unobserved confounders lead to violations of causal consistency (Def. 2). To witness, Fig. 9 shows observed trajectories in the offline data when the agent starts from state $s = (0, 0)$. When the weather is nice (`no-wind`) or the wind u is blowing towards `east` or `west`, the behavior policy selects action $x =$ `down`, similar to the interventional trajectories of Fig. 6. On the other hand, when the wind is blowing towards `north` or `south`, the behavior policy selects action $x =$ `right`, moving the agent towards the center of the board. Among all the possible next state in the observational data, we find that the agent will never reach the bottom left corner $s = (0, 2)$. This means that when evaluating the observational distribution $P(S_{t+1} \mid S_t, X_t)$, we must have

$$P(S_{t+1} = (0, 2) \mid S_t = (0, 0), X_t = (0, 1)) = 0 \tag{80}$$

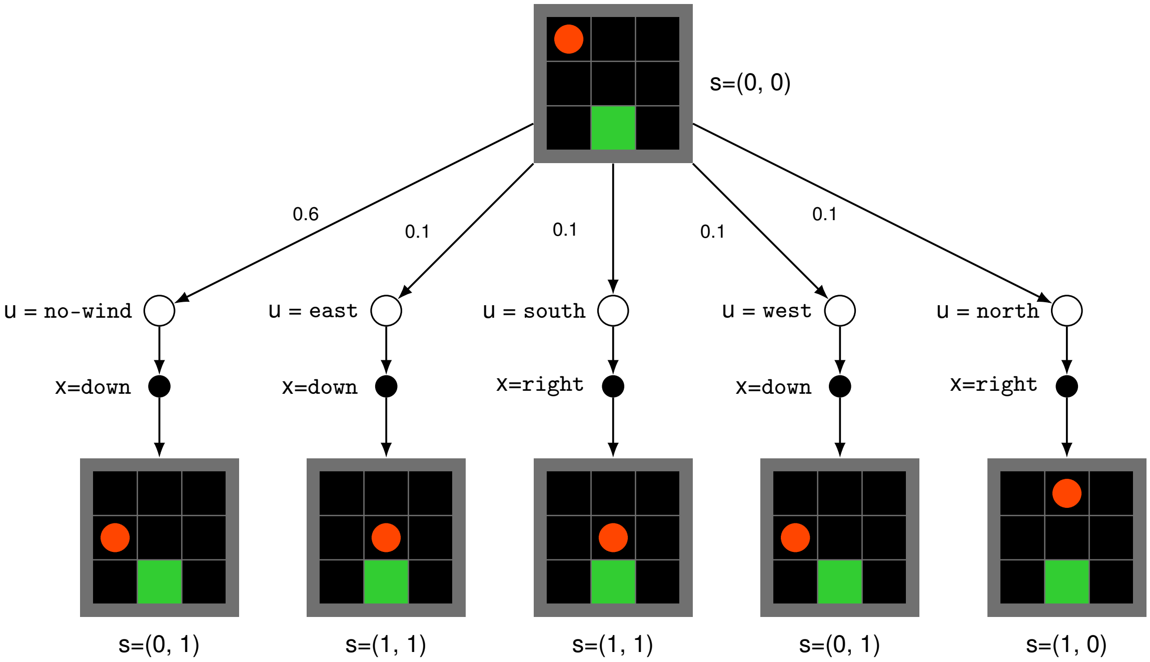


Figure 9: Trajectories sampled from the observational transition distribution $\widetilde{\mathcal{T}}$ induced by a confounded behavior policy f_X.

In other words, the nominal transition distribution $\widetilde{\mathcal{T}}(s, x, s') = 0$ when one observes the agent starting from $s = (0, 1)$, taking action $x = (0, 1)$, and moving to the next state $s' = (0, 2)$. Comparing the evaluations in Eqs. (79) and (80), we find that $P_{x_t}(s_{t+1} \mid s_t) \neq P(s_{t+1} \mid s_t, x_t)$, that is, causal consistency (Def. 2) does not hold between the agent's system dynamics in windy gridworld and the observational distribution generated by the confounded behavior policy in Fig. 8.

C Societal Impact

This paper investigates the theoretical framework of robust off-policy evaluation from biased offline data generated by a different behavior. Since unobserved confounding or no-overlap cannot be ruled out *a priori*, the agent's system dynamics in the environment cannot be fully identified from the offline data. To address this challenge, we proposed novel off-policy algorithms that allow the agent to derive informative bounds over value functions induced by a target policy from biased offline data. A positive impact of this work is that we address the potential risk of policy learning from offline data with the presence of unobserved confounding. Our framework is inherently robust against confounding bias and may apply to various consequential domains involving complex human interactions, including healthcare, marketing, finance, and autonomous driving. More broadly, automated decision systems using causal inference methods prioritize safety and robustness during their learning processes. Such requirements are increasingly essential since black-box AI systems are prevalent, and our understanding of their potential implications is still limited.