LOOK BACK: REINFORCEMENT LEARNING WITH COUNTERFACTUAL WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Agents in complex, stochastic environments must contend with dynamics that can exceed their world models' predictive capacity and randomness that can obscure the consequences of their actions. We hypothesize that, compared to imagining entirely new trajectories in such complex environments, it can be easier to transform observed trajectories with alternative actions. Counterfactual world models condition on observations to predict alternative outcomes under the same underlying environmental randomness, using experience as a reference for predicting action-dependent changes. We present a policy gradient estimator, REINFORCE Look Back (RLB), that uses counterfactual returns as baselines to distinguish action effects from environmental luck, and establish conditions for unbiasedness and variance reduction. In settings where environmental evolution is harder to predict than action-induced changes, we show that counterfactual world models are easier to learn and support better policy learning than ordinary world models. Our findings suggest that counterfactual thinking can help agents learn effective behavior without independently reproducing the full complexity of their worlds.

1 INTRODUCTION

Think of a commuter who arrives late after a morning of delayed trains, crowded platforms, and missed connections. To improve her future decisions, she could evaluate alternative departure times by imagining the resulting journeys and their outcomes (Sutton, 1991; Ha & Schmidhuber, 2018). Yet realistic imagination requires modeling unpredictable disruptions throughout a complex transportation system beyond her control. Moreover, random variations across independently simulated mornings can obscure whether leaving earlier would actually help.

This example illustrates two challenges for learning with world models in a big world (Javed & Sutton, 2024). World models allow agents to evaluate alternative actions without additional environment interaction, but their usefulness depends on both predictive accuracy and the informativeness of simulated outcomes. When environmental complexity exceeds an agent's modeling capacity, errors can compound across imagined trajectories (Janner et al., 2019; Assran et al., 2025). Even with an accurate model, environmental randomness can dominate simulated returns, obscuring action effects and increasing the variance of policy gradient estimates (Mao et al., 2018).

In this work, we study **counterfactual world models**, which condition on observed trajectories to predict alternative outcomes under the same underlying environmental randomness. Looking back, the commuter might ask: given how this morning unfolded, would leaving ten minutes earlier have helped? Answering this question may be easier since her actual experience provides a reference for predicting what would change under a different decision (Roese, 1997). A counterfactual model can use information about how the world unfolded and focus its predictive capacity on action-dependent changes (Buesing et al., 2018). Holding environmental randomness fixed can also help distinguish action effects from luck, reducing the variance of policy gradient estimates (Mesnard et al., 2020).

We explore two approaches to constructing counterfactual world models: (i) directly training models on paired trajectories generated under shared environmental randomness, and (ii) applying diffusion counterfactual inference (Sanchez & Tsaftaris, 2022) to trained diffusion world models (Alonso et al., 2024; Valevski et al., 2025). To use these models for reinforcement learning, we build on RLOO (Kool et al., 2019) and introduce **REINFORCE Look Back (RLB)**, a policy gradient algorithm that generates multiple counterfactual trajectories and uses their predicted returns as baselines.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

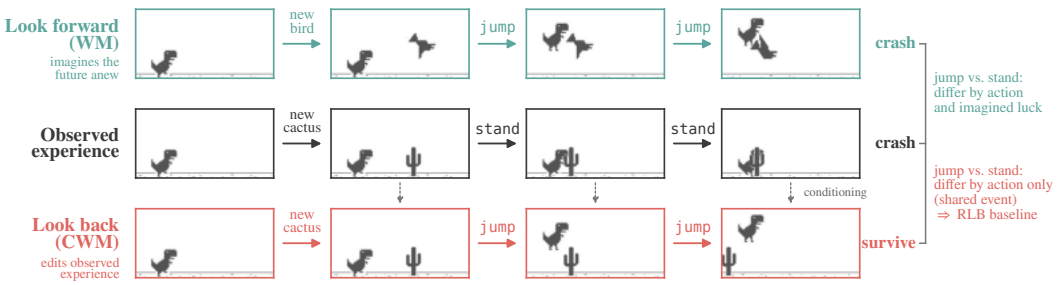

Figure 1: Overview. To evaluate an action `jump` against the observed action `stand`, a world model imagines the future anew, mixing the action's effect with luck (top), whereas a counterfactual world model edits the observed experience, isolating it (bottom).

Our main contributions are:

- We formalize counterfactual world models and two approaches to building them, covering settings with and without access to trajectory pairs generated under shared environmental randomness.
- We introduce RLB, a policy gradient algorithm based on counterfactual return baselines, and prove unbiasedness under exact counterfactuals and variance reduction under additional conditions.
- We show that counterfactual world models are easier to learn and yield better policies than ordinary world models when action effects are more predictable than environmental evolution.

Our results suggest that looking back at past experience can reduce the predictive burden of learning in complex environments and help agents more precisely identify the consequences of their actions.

2 Background

In this section, we provide an overview of the background on policy gradient methods for reinforcement learning and refer the reader to Sutton & Barto (2018) for further details.

Setup. We consider Markov decision processes (MDPs) with state $\mathbf{x} \in \mathbb{R}^d$, action $a \in A$ in finite A, initial state distribution μ, transition probability p, reward function r, finite horizon n, and discount factor $\gamma \in (0, 1]$. Given a policy π_θ parameterized with θ, where the environment is independent of θ, a trajectory $\tau := (\mathbf{x}_0, a_0, ..., \mathbf{x}_n)$ with state $\mathbf{x}_i$ and action a_i at step i is sampled as:

$$\mathbf{x}_0 \sim \mu, \qquad a_i \sim \pi_\theta(\cdot|\mathbf{x}_i), \qquad \mathbf{x}_{i+1} \sim p(\cdot|\mathbf{x}_i, a_i), \qquad i = 0, ..., n-1. \tag{1}$$

We write $\tau \sim p_\theta$ and denote the reward by $r_i := r(\mathbf{x}_i, a_i)$, and assume the usual regularity conditions for exchanging differentiation with expectation. Our goal is to optimize the policy to maximize the expected total return $J(\theta) := \mathbb{E}_{\tau \sim p_\theta}[G_0]$ where $G_i := \sum_{j=i}^{n-1} \gamma^{j-i} r_j$ is the return-to-go at step i.

Policy gradient methods. The policy can be optimized by updating θ with the gradient $\nabla_\theta J(\theta)$. Denoting the score of the policy at step i as $\mathbf{z}_i := \nabla_\theta \log \pi_\theta(a_i|\mathbf{x}_i)$, REINFORCE (Williams, 1992) estimates the policy gradient using $m \geq 1$ independently sampled trajectories:

$$\nabla_\theta J(\theta) = \mathbb{E}\left[\sum_{i=0}^{n-1} \gamma^i G_i \mathbf{z}_i\right] \approx \frac{1}{m} \sum_{j=1}^{m} \sum_{i=0}^{n-1} \gamma^i G_i^{(j)} \mathbf{z}_i^{(j)}, \qquad \tau^{(1)}, ..., \tau^{(m)} \overset{\text{iid}}{\sim} p_\theta, \tag{2}$$

where $G_i^{(j)}$ and $\mathbf{z}_i^{(j)}$ denote return-to-go and score of each trajectory $\tau^{(j)}$ at step i, respectively. While the estimation is unbiased, sampled returns can vary substantially, leading to a high variance. A common approach to reduce the variance without biasing the estimate is to use a possibly random baseline b_i that is independent of the action a_i given the state $\mathbf{x}_i$ (Greensmith et al., 2004):

$$\nabla_\theta J(\theta) = \mathbb{E}\left[\sum_{i=0}^{n-1} \gamma^i (G_i - b_i) \mathbf{z}_i\right], \qquad b_i \perp a_i | \mathbf{x}_i. \tag{3}$$

A standard choice is to fit a value function $b_i = \hat{V}_i(\mathbf{x}_i)$ to approximate the expected return $\mathbb{E}[G_i|\mathbf{x}_i]$. REINFORCE Leave-One-Out (RLOO) (Kool et al., 2019; Ahmadian et al., 2024) instead samples $m \geq 2$ trajectories at each step and uses leave-one-out returns to construct an empirical baseline:

$$\nabla_\theta J(\theta) \approx \frac{1}{m}\sum_{j=1}^{m}\sum_{i=0}^{n-1}\gamma^i(G_i^{(j)} - b_i^{(j)})\mathbf{z}_i^{(j)}, \quad b_i^{(j)} := \frac{1}{m-1}\sum_{\ell\neq j} G_i^{(\ell)}, \quad \tau_{\geq i}^{(1)}, ..., \tau_{\geq i}^{(m)} \overset{\text{iid}}{\sim} p_\theta(\cdot|\mathbf{x}_i), \tag{4}$$

where $G_i^{(j)}$ and $\mathbf{z}_i^{(j)}$ are return-to-go and score of each trajectory $\tau_{\geq i}^{(j)}$ at step i, and the trajectories at step i are sampled independently from the same state $\mathbf{x}_i \sim p_\theta$. To implement (4), one can sample a trajectory as root and then sample $m-1$ branching trajectories from every state, or a subset $B \subseteq \mathbb{R}^d$ to reduce cost (Schulman et al., 2015). Either way, this requires sampling new trajectories from an observed state. This can be done by resetting the environment, or by using a world model $p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i)$ (Ha & Schmidhuber, 2018; Xiao et al., 2025) with parameters ω learned via:

$$\mathcal{L}_{\mathsf{WM}}(\omega) := \sum_{i=0}^{n-1}\mathbb{E}_{(\mathbf{x}_i,a_i,\mathbf{x}_{i+1})\sim p_\phi}\left[-\log p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i,a_i)\right], \tag{5}$$

where a reference policy π_ϕ collects data $(\mathbf{x}_i, a_i, \mathbf{x}_{i+1}) \sim p_\phi$. In this way, policy gradient estimates can be improved by evaluating alternative actions using a world model. Yet, complex and stochastic environments, or "big worlds" (Javed & Sutton, 2024), remain challenging. When environmental complexity exceeds the capacity of the world model, its predictions would suffer from errors (Janner et al., 2019; Assran et al., 2025). Even with an accurate model, environmental randomness can dominate simulated returns, obscuring action effects and increasing the variance (Mao et al., 2018).

3 METHODS

In this section, we describe our ideas for remedying the aforementioned issues. In Section 3.1, we define counterfactual world models, show that they can be easier to learn than ordinary world models when environmental complexity outweighs actions, and outline two methods for their learning. In Section 3.2, we present a policy gradient estimator REINFORCE Look Back (RLB) via counterfactual world models, separating action effects from environmental luck to further reduce variance.

3.1 COUNTERFACTUAL WORLD MODELS

RLOO with a world model (4) predicts branching trajectories independently from the root. Our idea behind counterfactual world models is that conditioning on the whole root trajectory and predicting how it would become different under an alternative action can be easier, as it allows reusing observed environmental information. We formalize this intuition using structural causal models (Pearl, 2009).

Structural causal models (SCMs). Any MDP in the form of (1) can be expressed as a collection of deterministic functions that take noise variables (Buesing et al., 2018), defining an SCM:

$$\mathbf{x}_0 = \mu(\mathbf{u}_{-1}), \qquad \mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i, \mathbf{u}_i), \qquad a_i = \pi_\theta(\mathbf{x}_i, \boldsymbol{\epsilon}_i), \qquad i = 0, ..., n-1, \tag{6}$$

where the collection of environmental noise $\mathbf{u}$ and action noise $\boldsymbol{\epsilon}$ are mutually independent, and we abuse notations to write the functions underlying the initial distribution and the policy by μ and π_θ, respectively. Through the SCM formulation, all randomness in the MDP is captured by $(\mathbf{u}, \boldsymbol{\epsilon})$. There could be multiple SCMs consistent with a given MDP (Oberst & Sontag, 2019). We arbitrarily choose one of them, which suffices for unbiasedness of policy gradient estimation (Section 3.2).

Counterfactuals. Having expressed our MDP as an SCM, we can formalize counterfactual thinking after observing a transition $o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1})$. If we took an alternative action $a_i \sim \pi_\theta(\cdot|\mathbf{x}_i)$, and if all other factors in the environment unfolded identically, would the outcome become different? Under the SCM, this intuitively amounts to evaluating an alternative outcome $\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i, \mathbf{u}_i)$ where $\mathbf{u}_i$ is *the* environmental noise that has produced the factual observation $\hat{\mathbf{x}}_{i+1} = f(\hat{\mathbf{x}}_i, \hat{a}_i, \mathbf{u}_i)$. In general, $\mathbf{u}_i$ can be uncertain given o_i, and the counterfactual transition probability is (Pearl, 2009)

$$p(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i) = \int p(\mathbf{u}_i|o_i)\cdot\mathbf{1}(\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i, \mathbf{u}_i))\,\mathrm{d}\mathbf{u}_i, \tag{7}$$

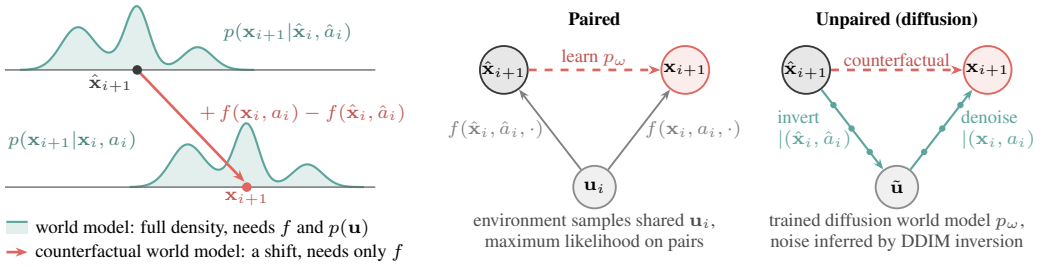

Figure 2: Counterfactual world models (CWMs). (Left) Predicting outcomes. A counterfactual is a shift of the factual outcome, requiring no model of the environmental noise. (Right) CWMs can be learned from paired outcomes with shared noise, or from unpaired data via diffusion inversion.

where $\mathbf{1}(\cdot)$ is delta function. While it is common to fix $\mathbf{x}_i = \hat{\mathbf{x}}_i$, we allow otherwise for convenience. We now show, using a simple additive model, that counterfactual prediction can be easier to model than ordinary transition. A proof is in Appendix B.1, and a visual illustration is in Figure 2.

Proposition 3.1. *Consider SCM transitions of the form $\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i) + \mathbf{u}_i$ where $\mathbf{u}_i \sim \nu$ is independent of the current state and action, and f and ν range over finite nonempty classes F and P, respectively, where every distribution in P has finite first moments. Then, measured by the number of distinct models, counterfactual prediction models have strictly lower complexity than ordinary transition kernels whenever $|P| > 1$.*

Counterfactual world models. Based on the SCM framework, we define a counterfactual world model as a model $p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)$ of the counterfactual probability (7) with parameters ω. It is distinguished from an ordinary world model $p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i)$, as it is additionally conditioned on factual observation $o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1})$ and predicts alternative outcome of taking an action a_i instead. Therefore, the relation between ordinary and counterfactual world models can be roughly understood as that between a generative model and an editing model.

In cases where paired observations with shared environmental noise can be obtained, a counterfactual world model can be learned with ordinary maximum likelihoods, using the following loss:

$$\mathcal{L}_{\mathsf{CWM}}(\omega) \coloneqq \sum_{i=0}^{n-1} \mathbb{E}_{\mathbf{x}_i, \hat{\mathbf{x}}_i, a_i, \hat{a}_i, \mathbf{u}_i} \left[-\log p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i) \right], \qquad o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1}), \tag{8}$$

where $\mathbf{x}_i, \hat{\mathbf{x}}_i \sim p_\phi$, $a_i \sim \pi_\phi(\cdot|\mathbf{x}_i)$ $\hat{a}_i \sim \pi_\phi(\cdot|\hat{\mathbf{x}}_i)$ are drawn with a reference policy π_ϕ, the environmental noise $\mathbf{u}_i$ is independently drawn from its SCM distribution, and the factual and counterfactual outcomes for training are obtained as $\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i, \mathbf{u}_i)$ and $\hat{\mathbf{x}}_{i+1} = f(\hat{\mathbf{x}}_i, \hat{a}_i, \mathbf{u}_i)$, respectively. We prove that, under mild assumptions, minimizing $\mathcal{L}_{\mathsf{CWM}}$ (8) recovers true counterfactuals:

Proposition 3.2. *Assume that, for every i:*

(i) The true and learned conditional distributions admit densities with respect to a common reference measure at each conditioning input. Their log densities, evaluated at the training target $\mathbf{x}_{i+1}$, are integrable under the training distribution.

(ii) There exists a choice of ω_* such that $p_{\omega_*}(\cdot|\mathbf{x}_i, a_i, o_i) = p(\cdot|\mathbf{x}_i, a_i, o_i)$ almost surely under the training distribution.

Then ω_* is a population minimizer of $\mathcal{L}_{\mathsf{CWM}}$, and every population minimizer recovers the true counterfactual conditional distribution almost surely under the training distribution.

A proof can be found in Appendix B.2. Notably, the result holds for any reference policy π_ϕ.

Diffusion counterfactual world models. The aforementioned approach requires access to paired factual and counterfactual observations, which can be unavailable for realistic environments.

In these cases, we propose to train an ordinary world model $p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i)$ based on state-action-conditioned diffusion models (Alonso et al., 2024; Valevski et al., 2025) to maximize the observation likelihood (5), then apply diffusion counterfactual inference (Sanchez & Tsaftaris, 2022) to turn it into a counterfactual world model $p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)$. This can be done, for each given observation

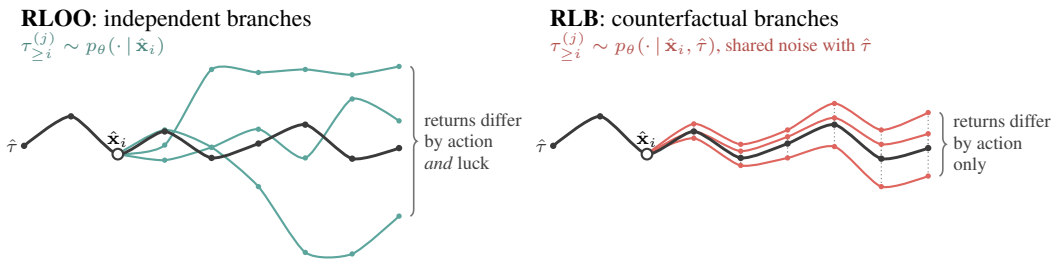

Figure 3: RLOO vs. RLB. Independent branches differ by both action and luck, whereas counterfactual branches share the factual noise and differ by action only.

$o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1})$, by first inferring the latent noise $\tilde{\mathbf{u}}$ underlying the transition $(\hat{\mathbf{x}}_i, \hat{a}_i) \mapsto \hat{\mathbf{x}}_{i+1}$ through diffusion noise inversion (Song et al., 2020), and then performing diffusion generation starting from the inferred noise $\tilde{\mathbf{u}}$ under an altered conditioning $(\mathbf{x}_i, a_i)$ to obtain $\mathbf{x}_{i+1}$ (Figure 2).

While this approach requires learning a world model, its prediction during inference is now counterfactual, which can be easier than ordinary world modeling as shown by Proposition 3.1. As such, this allows learning an approximate SCM and leveraging its counterfactuals from only observational data. This capability is useful for the policy gradient method we describe in the next subsection.

3.2 REINFORCE Look Back

We now present our policy gradient estimator. To motivate the method, we remark that the RLOO estimator (4) samples the root trajectory, here denoted $\hat{\tau}$, and branching trajectories, independently:

$$\tau^{(1)} = \hat{\tau} = (\hat{\mathbf{x}}_0, \hat{a}_0 ..., \hat{\mathbf{x}}_n) \sim p_\theta, \qquad \tau_{\geq i}^{(2)}, ..., \tau_{\geq i}^{(m)} \overset{\text{iid}}{\sim} p_\theta(\cdot | \hat{\mathbf{x}}_i). \tag{9}$$

While modeling the branches with a world model allows evaluating alternative actions, it can obscure action effects with environmental randomness. Our key idea is to replace the branching trajectories with counterfactuals relative to the root as the factual, thereby coupling environmental randomness. This leads to our policy gradient estimator, **REINFORCE Look Back (RLB)** (Figure 3):

$$\nabla_\theta J(\theta) \approx \frac{1}{m} \sum_{j=1}^{m} \sum_{i=0}^{n-1} \gamma^i (G_i^{(j)} - b_i^{(j)}) \mathbf{z}_i^{(j)}, \quad b_i^{(j)} := \frac{1}{m-1} \sum_{\ell \neq j} G_i^{(\ell)}, \quad \tau_{\geq i}^{(2)}, ..., \tau_{\geq i}^{(m)} \overset{\text{iid}}{\sim} p_\theta(\cdot | \hat{\mathbf{x}}_i, \hat{\tau}) \tag{10}$$

where we define the trajectory-level counterfactual probability $p_\theta(\tau_{\geq i} | \hat{\mathbf{x}}_i, \hat{\tau})$ as follows:

$$p_\theta(\tau_{\geq i} | \mathbf{x}_i, \hat{\tau}) := \prod_{k=i}^{n-1} \pi_\theta(a_k | \mathbf{x}_k) \cdot p(\mathbf{x}_{k+1} | \mathbf{x}_k, a_k, o_k), \quad o_k = (\hat{\mathbf{x}}_k, \hat{a}_k, \hat{\mathbf{x}}_{k+1}), \tag{11}$$

and model $p(\mathbf{x}_{k+1} | \mathbf{x}_k, a_k, o_k)$ using a counterfactual world model p_ω (Section 3.1). By coupling environmental randomness as much as possible across trajectories, much of the sampled return can be attributed to the specific action taken, potentially improving the policy gradient estimate. As per RLOO, we can infer the $m-1$ counterfactual trajectories from every state of the factual, or a subset $B \subseteq \mathbb{R}^d$ to reduce computational cost. We now prove that RLB is unbiased under proper conditions:

Proposition 3.3. *Assume that all noise variables in the SCM (6) are mutually independent and have distributions independent of θ. Let $m \geq 2$ and $\hat{\tau} \sim p_\theta$. At every step i, sample the counterfactual trajectories as in (10), using exact counterfactual transition distributions and fresh independent policy randomness. Under the usual regularity and integrability conditions, RLB is unbiased.*

A proof is in Appendix B.3. We now prove that if environmental randomness outweighs action, RLB can reduce variance over RLOO and REINFORCE. A proof is in Appendix B.4.

Proposition 3.4. *Consider an MDP with a fixed initial state $\mathbf{x}_0$, $n = 2$, $\gamma = 1$, and $\mathbf{x}_1 = f(\mathbf{x}_0, a_0) + \mathbf{u}_0$. Suppose $r_0 = 0$, the action at step 1 is fixed and independent of θ, and $G_0 = r_1 = \mathbf{c}^\top \mathbf{x}_1$ for a fixed vector $\mathbf{c}$. Assume f and the noise distribution are independent of θ, $\mathbf{u}_0$ is independent of a_0, and $\mathbb{E}[\|\mathbf{u}_0\|^2] < \infty$. Assume the initial policy has differentiable, positive probabilities on the finite action set A, and write $\mathbf{z}_0 = \nabla_\theta \log \pi_\theta(a_0 | \mathbf{x}_0)$. For $m \geq 2$,*

REINFORCE and RLOO use m independent ordinary trajectories. RLB uses one factual trajectory and $m-1$ exact counterfactuals of the stated SCM, with fresh independent policy samples. Define $\widehat{\mathbf{g}}_{\mathsf{RF}} = \frac{1}{m}\sum_{j=1}^{m} G_0^{(j)}\mathbf{z}_0^{(j)}$ and, for each of RLOO and RLB, define, using the returns from the respective sampling scheme, $\widehat{\mathbf{g}} = \frac{1}{m}\sum_{j=1}^{m}(G_0^{(j)} - b_0^{(j)})\mathbf{z}_0^{(j)}$ where $b_0^{(j)} = \frac{1}{m-1}\sum_{\ell\neq j} G_0^{(\ell)}$. Let $\sigma^2 = \mathrm{Var}(\mathbf{c}^\top \mathbf{u}_0)$ and $L = \max_{a\in A} |\mathbf{c}^\top f(\mathbf{x}_0, a) - \mathbb{E}[\mathbf{c}^\top f(\mathbf{x}_0, a_0)]|$. Then:

(i) RLB reduces covariance relative to RLOO: $\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLOO}}) - \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) = \frac{\sigma^2}{m-1}\mathbb{E}[\mathbf{z}_0\mathbf{z}_0^\top] \succeq 0$. *The trace reduction is strict if $\sigma^2 > 0$ and $\mathbb{E}[\|\mathbf{z}_0\|^2] > 0$.*

(ii) If also $\sigma^2 \geq \frac{m}{m-1}L^2$, RLB reduces covariance over REINFORCE: $\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) \preceq \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RF}})$.

4 Related Work

We provide an overview of related work. An extended discussion can be found in Appendix A.

World models. In model-based reinforcement learning, world models support policy learning through imagined experience (Sutton, 1991; Ha & Schmidhuber, 2018). Yet, model errors arising from limited capacity can limit their utility (Janner et al., 2019), especially when modeling a big world (Javed & Sutton, 2024). Prior work has addressed this challenge by directing model capacity toward predictions needed for value estimation and planning (Grimm et al., 2020; LeCun, 2022). We study a complementary approach of using observed trajectories as conditioning information when predicting the consequences of alternative actions. This can simplify prediction when environmental evolution is difficult to model but action effects are comparatively simple (Buesing et al., 2018).

Counterfactual reinforcement learning. Prior work has used counterfactual reasoning (Roese, 1997) to improve reinforcement learning methods. Buesing et al. (2018) show that conditioning simulations on logged experience can improve policy evaluation and search. CCA (Mesnard et al., 2020) learns value baselines that estimate counterfactual returns in order to better distinguish action effects from environmental luck, adding an action-independence loss to encourage unbiasedness. Our approach learns imaginations using a counterfactual world model to construct policy gradient baselines, harnessing benefits of counterfactuals in both model learning and credit assignment.

Policy gradient baselines. Our policy gradient estimator builds on prior work on baselines, which can reduce variance without biasing the gradient under proper conditions (Greensmith et al., 2004). RLOO (Kool et al., 2019) constructs baselines from independent trajectories from the same initial state, and world models can provide these trajectories when the environment cannot be reset (Xiao et al., 2025). For environments with simulators, prior work has shown that constructing baselines from trajectories with shared randomness can further reduce variance (Schulman et al., 2015; Mao et al., 2018; Bazerghi et al., 2025) through a common random number effect. Our approach constructs such baselines using counterfactual world models without requiring direct simulator access.

Counterfactual modeling. Counterfactual modeling builds upon the causal inference framework (Pearl, 2009), with approaches including learning from paired observations with shared underlying randomness (Baradel et al., 2019) and learning a diffusion model to generate counterfactuals from noise inversion of observed data (Sanchez & Tsaftaris, 2022). We adapt both to world modeling, using trajectory pairs generated under different actions and shared environmental randomness, or extending diffusion counterfactuals to action-conditioned prediction. The latter fits naturally with model-based reinforcement learning, as it builds upon diffusion world models trained on interaction data (Alonso et al., 2024; Valevski et al., 2025) without requiring paired counterfactual supervision.

5 Experiments

Through our experiments, we aim to answer the following key questions:

Q1. When using ground-truth counterfactuals, does RLB yield better policies than RLOO that uses ground-truth environmental simulations, as well as other policy gradient methods?

Q2. Are counterfactual world models easier to learn, and do they yield better policies via RLB than ordinary world models when action effects are more predictable than environmental evolution?

Q3. Can diffusion world models trained with observational data be turned into counterfactual world models and support effective policy learning through RLB?

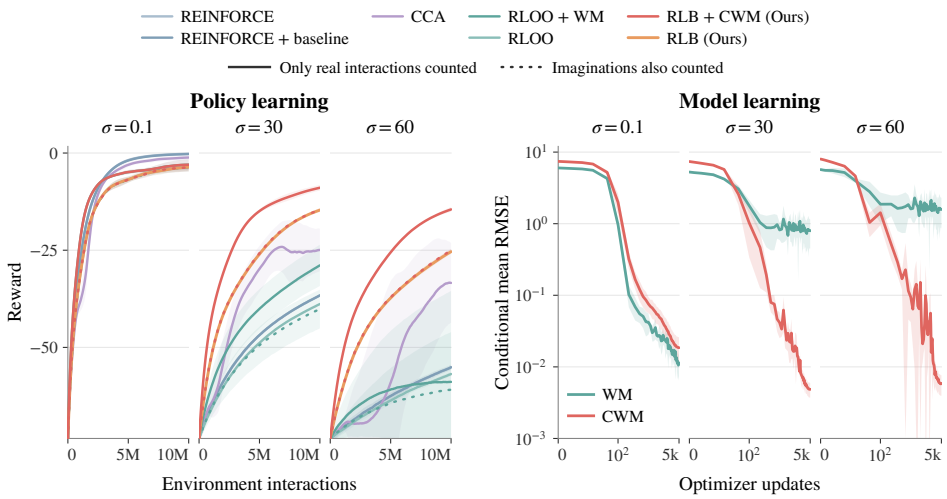

Figure 4: Results for the contextual bandit problem across six independent randomized trials. (Left) Policy learning across various levels of environmental noise σ. (Right) Prediction error of ordinary and counterfactual world models over training steps. *For both, the minimum attainable value is zero.*

5.1 Contextual bandit

We first study the benefits of counterfactuals in a setting where the optimal policy is known.

Setup. We consider a contextual bandit problem, adapted from Mesnard et al. (2020). The agent receives a context $\mathbf{c} \in \{-10, ..., 10\}$, and based on the context, selects an action $a \in \{-10, ..., 10\}$. Then, based on an environmental noise $\mathbf{u} \sim \mathsf{N}(0, \sigma^2)$ with noise level $\sigma > 0$, the agent receives a reward $r = \sigma^2 - (\mathbf{c} - (a + \mathbf{u}))^2$. This decision process can be expressed as an SCM of the form (6):

$$\mathbf{c} \sim \mathsf{U}(\{-10, ..., 10\}), \qquad \mathbf{x} = (\mathbf{c}, a + \mathbf{u}) = f(\mathbf{c}, a, \mathbf{u}), \qquad a = \pi_\theta(\mathbf{c}, \boldsymbol{\epsilon}). \tag{12}$$

For this problem, the optimal policy is choosing $a = \mathbf{c}$, which yields an average reward of zero. Yet, the reward signal is contaminated by the environmental noise $\mathbf{u}$ which is uncorrelated to the action. The higher the noise level σ is, the more difficult it becomes to clarify the effect of an action, since high rewards could be due to a fortunate draw $\mathbf{u} \approx a - \mathbf{c}$ instead of a proper choice of action.

In this setting, we evaluate REINFORCE, REINFORCE with a learned value baseline, RLOO, and the proposed RLB. We additionally test CCA (Mesnard et al., 2020), a method that learns a hindsight value baseline and is specialized for settings with high environmental uncertainty. For RLOO and RLB, we use $m = 4$ branches, and test both branching with ground-truth transitions and counterfactuals, and also with a world model $p(\mathbf{x}|\mathbf{c}, a)$ or with a counterfactual world model $p(\mathbf{x}|\mathbf{c}, a, o)$ conditioned on factual observations $o = (\hat{\mathbf{c}}, \hat{a}, \hat{\mathbf{x}})$, trained with maximum likelihoods.

Results. Figure 4 shows the results. As stochasticity in the environment increases, REINFORCE and RLOO degrade rapidly, presumably due to the difficulty of isolating action effects from environmental luck. Although CCA attains better performance, it exhibits substantial variation across runs, which we conjecture is caused by optimization instabilities from having to balance policy learning and auxiliary independence regularizer. In contrast, RLB with ground-truth counterfactuals can separate action effects without using an additional loss, converging faster and in a more stable manner than the baselines as noise increases (**Q1**). Furthermore, learning a counterfactual world model saves $m - 1 = 3$ environment interactions out of $m = 4$, attaining the best result. In addition, we find that as environmental noise becomes larger, it becomes easier to learn counterfactual world models than ordinary ones. This is reflected in policy learning, as when compared at matched environment interactions and model imaginations, RLB with counterfactual model matches using true counterfactuals, whereas RLOO with world model lags behind using true simulations (**Q2**).

5.2 Multi-agent bandit

In the prior experiment, difficulties in policy and model learning primarily arose from large environmental noise with a relatively simple distribution. We now consider a setting where complexity

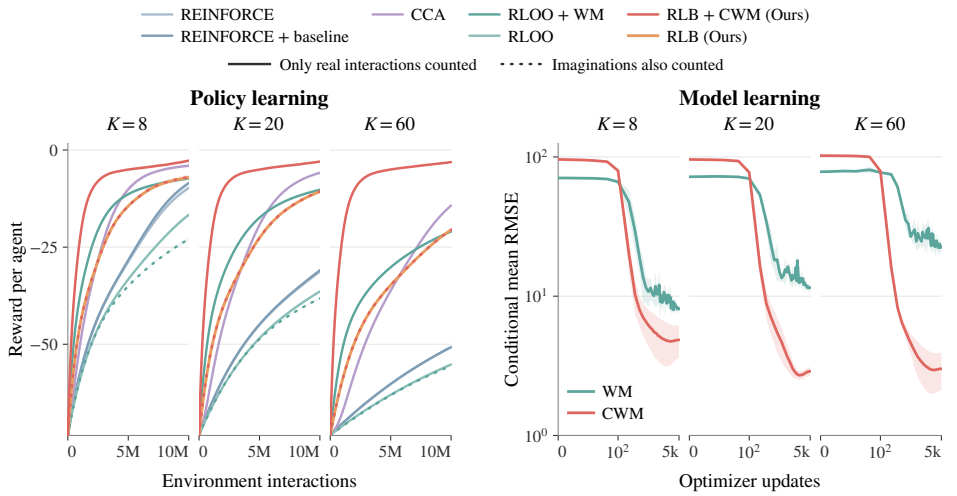

Figure 5: Results for the multi-agent bandit problem across six randomized trials. (Left) Policy learning across various numbers K of agents in the environment. (Right) Prediction error of ordinary and counterfactual world models over training steps. *For both, the minimum attainable value is zero.*

of the environment exceeds that of learners. We achieve this through a multi-agent setting where each agent faces a strictly bigger environment than itself in terms of computational capacity, because there are multiple other agents of equal capacity (Javed & Sutton, 2024), as illustrated in Figure 6.

Setup. For the environment, we employ $K > 1$ copies of the previous contextual bandit (Mesnard et al., 2020). Each of the K agents interacts with its own version of the environment, but rewards are now tied across agents. The i-th agent receives a context $\mathbf{c}^i \sim \mathsf{U}(\{-10, ..., 10\})$ and selects an action $a^i \in \{-10, ..., 10\}$, where $\mathbf{c}$ and a now denote the concatenation of all agents' contexts and actions. Then, every agent receives reward $r = -\sum_i (\mathbf{c}^i - a^i)^2$. We write this as an SCM (6):

$$\mathbf{x}^i = \sum_j (\mathbf{c}^j - a^j)^2 = f^i(\mathbf{c}^i, a^i, \mathbf{u}^i), \qquad \mathbf{u}^i = (\mathbf{c}^j, \boldsymbol{\epsilon}^j)_{j \neq i}, \qquad a^i = \pi_{\theta^i}(\mathbf{c}^i, \boldsymbol{\epsilon}^i), \tag{13}$$

where $\mathbf{x}^i$, f^i, π_{θ^i} denote the state, transition, and policy for the i-th agent, respectively. The optimal policy for every agent is choosing $a^i = \mathbf{c}^i$, yielding an average reward of zero for all agents. Yet, the reward signal is corrupted by other agents' contexts and actions, constituting the environmental noise $\mathbf{u}^i$. As the number of interacting agents increases, it becomes harder for each agent to understand the effect of its action, because high or low rewards could be due to other agents' decisions.

We test RLB along with baselines from Section 5.1. For RLOO and RLB with true simulations, there is one shared environmental interaction across the agents, and $m - 1$ branching simulations for each of the K agents, where we use $m = 4$. This makes them costly, as each policy gradient estimation costs $1 + K(m - 1)$ environmental interactions. If each agent has its own world model $p(\mathbf{x}^i|\mathbf{c}^i, a^i)$ or a counterfactual world model $p(\mathbf{x}^i|\mathbf{c}^i, a^i, o^i)$ with $o^i = (\hat{\mathbf{c}}^i, \hat{a}^i, \hat{\mathbf{x}}^i)$, this can be substantially reduced to a single interaction by replacing the rest with imaginations. However, world model of each agent needs to predict the return contribution of all other agents, while the counterfactual one can condition on their observed contributions. We thus expect counterfactual modeling to be easier (Figure 6).

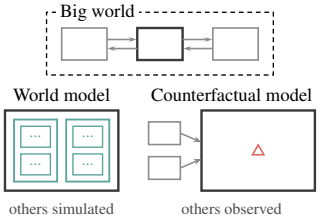

Figure 6: An agent among others of equal capacity (top). A world model must simulate them (left). A counterfactual world model observes them (right).

Results. Figure 5 shows the results. As complexity of the environment increases, standard policy gradient methods degrade rapidly, and RLB with true counterfactuals overall outperform RLOO with true simulations (**Q1**). While RLB with true counterfactuals is outperformed by CCA at matched interactions due to its high cost, we find that counterfactual world models can be learned well and easier than ordinary ones even with a large number of other agents, thereby substantially improving data efficiency of RLB and outperforming CCA at matched real interactions (**Q2**).

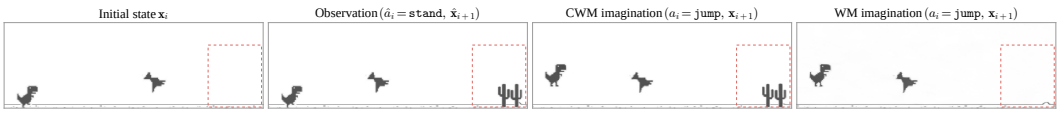

Figure 7: Visualization of an environmental transition under action `stand`, a counterfactual world model imagination under action `jump`, and a world model imagination under action `jump`.

Table 1: Model imaginations assessed following Melistas et al. (2024).

Criterion	CWM	WM
Composition (ℓ_1) ↓	**0.37**	1.89
Effectiveness (macro-F1) ↑	**0.999**	0.993
Realism (FID) ↓	**15.94**	37.72
Minimality (CLD) ↓	**0.296**	0.327
Minimality (obstacle ℓ_1) ↓	**2.34**	9.57

Table 2: Results for policy learning across three independent randomized training.

Method	Collision (%) ↓	Survival ↑
REINFORCE	15.4 ± 14.5	56 ± 30
REINFORCE + baseline	4.6 ± 2.1	98 ± 28
RLOO	3.4 ± 0.5	113 ± 13
RLOO + WM	4.5 ± 2.7	103 ± 32
RLB (Ours)	3.0 ± 0.4	118 ± 9
RLB + CWM (Ours)	$\mathbf{2.6 \pm 0.1}$	$\mathbf{129 \pm 1}$

5.3 The dinosaur game

The preceding experiments show the benefits of counterfactual modeling, but the models therein are trained with paired data with coupled environmental noise, which might be unavailable in practice. Therefore, we now test whether a counterfactual world model can be obtained by training a diffusion world model and employing diffusion counterfactual inference, as proposed in Section 3.1.

Setup. We use Chrome Dino (Google, 2014), a platform game with a known method to train diffusion world models. At each timestep, the agent receives 128×64 image as an observation $\mathbf{x}_i$ and chooses an action $a_i \in \{\texttt{stand}, \texttt{jump}, \texttt{duck}\}$. The goal is to survive by avoiding incoming obstacles including cacti on the ground and birds at varying heights, and the agent gets the reward $r_i = +1$ if it passes an obstacle and -1 if it collides. Since the obstacles are generated randomly independent of the action, we can view their generation as the environmental noise $\mathbf{u}_i$ that obscures action effect. We train a diffusion world model based on the protocol of Valevski et al. (2025) and turn it into a counterfactual world model using diffusion counterfactual inference (Sanchez & Tsaftaris, 2022) which was originally tested only on class-conditional image generation. Further details are in Appendix C. We evaluate model imaginations following the criteria proposed in Melistas et al. (2024), measuring composition, effectiveness, realism, and minimality (see Appendix C for details). We use $m = 4$ for RLB and RLOO training, and evaluate each learned policy by running 100 test episodes and measuring collision rate per obstacle and mean survival time before the first collision.

Results. Figure 7 and Table 1 demonstrate the quality of counterfactual world model imaginations. In addition to CLD, we measure minimality by the pixel ℓ_1 error between factual and imagined obstacles within the red regions in Figure 7. We find that the counterfactual predictions preserve environmental randomness given in the factual observation while incorporating the effect of an alternative action, thereby attaining good quantitative metrics of being a proper counterfactual. This supports that diffusion world models trained with observational data can be turned into counterfactual world models (**Q3**). Table 2 shows the results for downstream policy learning. Using either true simulations or model imaginations, RLB overall outperforms other methods, supporting that diffusion counterfactual modeling supports effective policy learning (**Q1**–**Q3**). For RLB, we find that model imaginations yield a better policy than true simulations, implying that diffusion counterfactuals offer a favorable bias-variance tradeoff.

6 Conclusion

We studied counterfactual world models, which predict how observed trajectories would change under alternative actions. Our REINFORCE Look Back (RLB) uses their predicted returns as baselines to separate action effects from environmental luck. It is unbiased under exact counterfactuals and can reduce variance when environmental noise dominates. When action effects were more predictable than the environment, counterfactual world models were easier to learn and yielded better policies than ordinary ones. Limitations include the need for future work on larger diffusion models and connecting to popular methods like GRPO. Our results suggest that agents in big worlds may learn by looking back at what happened, without modeling the full complexity of their environments.

AI USE STATEMENT

We used generative AI tools to assist with writing, searching relevant literature, implementation and debugging, and checking mathematical derivations. The authors take full responsibility for the final manuscript and accompanying materials.

ETHICS STATEMENT

We do not anticipate ethical concerns specific to our work beyond those generally associated with generative world models and small-scale reinforcement learning.

REPRODUCIBILITY STATEMENT

We detail the implementation and experimental settings in Appendix C. We plan to release the code and checkpoints for reproducing our experiments upon acceptance.

REFERENCES

Pieter Abbeel, Morgan Quigley, and Andrew Y Ng. Using inaccurate models in reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pp. 1–8, 2006. (page 13)

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, 2024. (pages 3, 13)

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024. (pages 1, 4, 6)

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. (pages 1, 3)

Fabien Baradel, Natalia Neverova, Julien Mille, Greg Mori, and Christian Wolf. Cophy: Counterfactual learning of physical dynamics. *arXiv preprint arXiv:1909.12000*, 2019. (pages 6, 13)

Audrey Bazerghi, Sébastien Martin, and Garrett van Ryzin. Relative monte carlo for reinforcement learning, 2025. (pages 6, 13)

Lars Buesing, Theophane Weber, Yori Zwols, Sebastien Racaniere, Arthur Guez, Jean-Baptiste Lespiau, and Nicolas Heess. Woulda, coulda, shoulda: Counterfactually-guided policy search. *arXiv preprint arXiv:1811.06272*, 2018. (pages 1, 3, 6, 13)

Patrick Chao, Patrick Blöbaum, Sapan Patel, and Shiva Prasad Kasiviswanathan. Modeling causal mechanisms with diffusion models for interventional and counterfactual queries. *arXiv preprint arXiv:2302.00860*, 2023. (page 13)

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. (page 13)

Google. Chrome dino (t-rex runner). `chrome://dino`, 2014. Accessed: 2026-09-24. (page 9)

Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(Nov):1471–1530, 2004. (pages 2, 6)

Christopher Grimm, André Barreto, Satinder Singh, and David Silver. The value equivalence principle for model-based reinforcement learning. *Advances in neural information processing systems*, 33:5541–5552, 2020. (page 6)

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3):440, 2018. (pages 1, 3, 6)

Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado P van Hasselt, Gregory Wayne, Satinder Singh, Doina Precup, et al. Hindsight credit assignment. *Advances in neural information processing systems*, 32, 2019. (page 13)

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. (page 19)

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019. (pages 1, 3, 6)

Steeven Janny, Fabien Baradel, Natalia Neverova, Madiha Nadri, Greg Mori, and Christian Wolf. Filtered-cophy: Unsupervised learning of counterfactual physics in pixel space. *arXiv preprint arXiv:2202.00368*, 2022. (page 13)

Khurram Javed and Richard S Sutton. The big world hypothesis and its ramifications for artificial intelligence. In *Finding the Frame: An RLC Workshop for Examining Conceptual Frameworks*, 2024. (pages 1, 3, 6, 8)

Wouter Kool, Herke Van Hoof, and Max Welling. Buy 4 reinforce samples, get a baseline for free!, 2019. (pages 1, 3, 6, 13)

Qingfeng Lan. From REINFORCE to dr. GRPO: A unified perspective on LLM post-training. In *The Blogpost Track at ICLR 2026*, 2026. URL `https://iclr-blogposts.github.io/2026/blog/2026/llm-post-training/`. (page 13)

Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022. (page 6)

Hongzi Mao, Shaileshh Bojja Venkatakrishnan, Malte Schwarzkopf, and Mohammad Alizadeh. Variance reduction for reinforcement learning in input-driven environments. *arXiv preprint arXiv:1807.02264*, 2018. (pages 1, 3, 6, 13)

Thomas Melistas, Nikos Spyrou, Nefeli Gkouti, Pedro Sanchez, Athanasios Vlontzos, Yannis Panagakis, Giorgos Papanastasiou, and Sotirios A Tsaftaris. Benchmarking counterfactual image generation. *Advances in Neural Information Processing Systems*, 37:133207–133230, 2024. (pages 9, 20)

Thomas Mesnard, Théophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Tom Stepleton, Nicolas Heess, Arthur Guez, et al. Counterfactual credit assignment in model-free reinforcement learning. *arXiv preprint arXiv:2011.09464*, 2020. (pages 1, 6, 7, 8, 13, 18)

Thomas Mesnard, Wenqi Chen, Alaa Saade, Yunhao Tang, Mark Rowland, Theophane Weber, Clare Lyle, Audrunas Gruslys, Michal Valko, Will Dabney, et al. Quantile credit assignment. In *International Conference on Machine Learning*, pp. 24517–24531. PMLR, 2023. (page 13)

Michael Oberst and David Sontag. Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pp. 4881–4890. PMLR, 2019. (pages 3, 13)

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition, 2009. (pages 3, 6)

Neal J Roese. Counterfactual thinking. *Psychological bulletin*, 121(1):133, 1997. (pages 1, 6)

Pedro Sanchez and Sotirios A Tsaftaris. Diffusion causal models for counterfactual estimation. *arXiv preprint arXiv:2202.10166*, 2022. (pages 1, 4, 6, 9, 13)

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. Pmlr, 2015. (pages 3, 6, 13)

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. (page 13)

Yiqing Shen, Aiza Maksutova, Chenjia Li, and Mathias Unberath. Counterfactual world models via digital twin-conditioned video diffusion. *arXiv preprint arXiv:2511.17481*, 2025. (page 13)

Sean R Sinclair, Felipe Vieira Frujeri, Ching-An Cheng, Luke Marshall, Hugo De Oliveira Barbalho, Jingling Li, Jennifer Neville, Ishai Menache, and Adith Swaminathan. Hindsight learning for mdps with exogenous inputs. In *International Conference on Machine Learning*, pp. 31877–31914. PMLR, 2023. (page 13)

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. (page 5)

Richard S Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991. (pages 1, 6)

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. (page 2)

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations*, volume 2025, pp. 73754–73776, 2025. (pages 1, 4, 6, 9)

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992. (page 2)

Junjin Xiao, Yandan Yang, Xinyuan Chang, Ronghan Chen, Feng Xiong, Mu Xu, Wei-Shi Zheng, and Qing Zhang. World-env: Leveraging world model as a virtual environment for vla post-training. *arXiv preprint arXiv:2509.24948*, 2025. (pages 3, 6)

A Extended related work

In this section, we provide an extended discussion of related work across domains, including on causal reinforcement learning and variance reduction using common random number effect.

Policy gradients with grouped baselines. RLB constructs learning signals by comparing multiple samples associated with a common input. Its parent method, RLOO (Kool et al., 2019; Ahmadian et al., 2024) uses the mean return of other independently sampled trajectories as a leave-one-out baseline. Relatedly, GRPO (Shao et al., 2024) normalizes rewards within groups of responses to the same prompt for clipped policy updates. Yet, sharing a starting state or prompt does not itself couple environmental randomness. RLB adapts RLOO baseline but generates counterfactual branches conditioned on a factual. Given the close relationship between RLOO and GRPO (Lan, 2026), we conjecture that our approach could be extended to GRPO, which we leave as future work.

Policy gradients with shared randomness. Our work is closely related with prior work that leverages common random numbers to reduce the variance of policy gradient estimators. TRPO's "vine" procedure (Schulman et al., 2015) compares actions through rollouts using common random numbers and requires simulator resets to sampled states. Input-dependent baselines (Mao et al., 2018) condition value predictions on observed future exogenous inputs. Hindsight learning (Sinclair et al., 2023) uses logged exogenous inputs and known dynamics to evaluate actions through hindsight planning. rMC (Bazerghi et al., 2025) estimates policy gradients for finite MDPs from return differences between a root trajectory and coupled true counterfactuals for all alternative actions run until they merge with the root. These methods exploit simulator access or explicitly available exogenous inputs. RLB obtains coupled rollouts through learned counterfactual dynamics, allowing policy learning without online simulator resets or direct online observation of environmental noise.

Hindsight and counterfactual credit assignment. Prior work has proposed to leverage hindsight and counterfactual predictions to distinguish action effects from environmental luck. HCA (Harutyunyan et al., 2019) assigns credit using action probabilities conditioned on future states or returns. CCA (Mesnard et al., 2020) learns value baselines from features of future trajectories, encouraging their conditional independence from current action. QCA (Mesnard et al., 2023) compares action-conditioned return distributions at the quantile level and uses a hindsight predictor to infer it from future observations. COMA (Foerster et al., 2018) uses a centralized critic to marginalize one agent's action while fixing the others' actions. RLB constructs baselines from generated counterfactual trajectories, without a separate critic that predicts returns. In particular, it couples transition outcomes through factual observations, instead of, for instance, using return quantiles.

Counterfactual inference for policy search and evaluation. Similar to our counterfactual world models, prior work has used observed experience to improve simulations of alternative decisions. Abbeel et al. (2006) correct approximate dynamics using real trajectories for local policy improvement in nearly deterministic settings. CF-GPS (Buesing et al., 2018) infers scenarios from logged episodes, simulates alternative policies, and improves policies through return-weighted regression. It motivates factual conditioning when scenario distributions are harder to model than causal mechanisms, thereby inspiring our approach. Oberst & Sontag (2019) reuse inferred Gumbel noise under alternative policies for counterfactual off-policy evaluation. Distinguished from these approaches, we employ a grouped policy gradient estimator using learned counterfactual transitions, where both the factual and counterfactual contribute to each update.

Learned and generative counterfactual inference. RLB is related to, and can benefit from, methods for inferring and learning counterfactual dynamics. CoPhy (Baradel et al., 2019) predicts physical trajectories under altered initial conditions from factual experience, using paired factual and counterfactual training data similarly to our counterfactual world models. Filtered-CoPhy (Janny et al., 2022) extends this to video prediction without object position or physical property supervision, using paired videos. We study a similar prediction problem in action-conditioned prediction setting, and further study modeling difficulty depending on environmental properties. Diff-SCM (Sanchez & Tsaftaris, 2022) and DCM (Chao et al., 2023) use diffusion-based latent inference to generate counterfactuals in class-conditional image generation and a medical application, respectively. CWMDT (Shen et al., 2025) conditions video diffusion on structured scene descriptions modified by a language model. Distinct from these works, we apply diffusion counterfactual inference to an action-conditioned world model trained on interaction data.

B PROOFS

B.1 PROOF OF PROPOSITION 3.1

Proposition 3.1. *Consider SCM transitions of the form $\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i) + \mathbf{u}_i$ where $\mathbf{u}_i \sim \nu$ is independent of the current state and action, and f and ν range over finite nonempty classes F and P, respectively, where every distribution in P has finite first moments. Then, measured by the number of distinct models, counterfactual prediction models have strictly lower complexity than ordinary transition kernels whenever $|P| > 1$.*

Proof. First, consider a factual observation $o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1})$. Since $\hat{\mathbf{x}}_{i+1} = f(\hat{\mathbf{x}}_i, \hat{a}_i) + \mathbf{u}_i$, the observation determines the environmental noise:

$$\mathbf{u}_i = \hat{\mathbf{x}}_{i+1} - f(\hat{\mathbf{x}}_i, \hat{a}_i).$$

Substituting this into the queried transition gives

$$\mathbf{x}_{i+1} = \hat{\mathbf{x}}_{i+1} + f(\mathbf{x}_i, a_i) - f(\hat{\mathbf{x}}_i, \hat{a}_i).$$

Thus the counterfactual prediction map depends on f but not on ν. Let C be the family of distinct maps given by this formula, and let W be the family of distinct ordinary transition kernels induced by all $(f, \nu) \in F \times P$.

Next, we show that the same ordinary kernel cannot correspond to two different counterfactual maps. Suppose (f, ν) and $(\tilde{f}, \tilde{\nu})$ induce the same ordinary transition kernel. Their conditional means must agree, so

$$f(\mathbf{x}, a) + \mathbb{E}_{\mathbf{u}\sim\nu}[\mathbf{u}] = \tilde{f}(\mathbf{x}, a) + \mathbb{E}_{\mathbf{u}\sim\tilde{\nu}}[\mathbf{u}]$$

for every $(\mathbf{x}, a)$. These expectations exist by the finite first moment assumption. Subtracting the corresponding equality at $(\hat{\mathbf{x}}, \hat{a})$ gives

$$f(\mathbf{x}, a) - f(\hat{\mathbf{x}}, \hat{a}) = \tilde{f}(\mathbf{x}, a) - \tilde{f}(\hat{\mathbf{x}}, \hat{a}).$$

Therefore the two counterfactual maps are identical.

Finally, choose one representative $f \in F$ for each distinct map in C. For this fixed f, varying ν over P produces $|P|$ distinct ordinary kernels: at any fixed $(\mathbf{x}, a)$, translation by $f(\mathbf{x}, a)$ preserves distinctness of probability distributions. By the preceding argument, kernels obtained from representatives of different counterfactual maps cannot coincide. Consequently,

$$|W| \geq |P|\,|C|.$$

Since C is nonempty, $|P| > 1$ implies $|W| > |C|$. □

B.2 PROOF OF PROPOSITION 3.2

Proposition 3.2. *Assume that, for every i:*

(i) The true and learned conditional distributions admit densities with respect to a common reference measure at each conditioning input. Their log densities, evaluated at the training target $\mathbf{x}_{i+1}$, are integrable under the training distribution.

(ii) There exists a choice of ω_* such that $p_{\omega_*}(\cdot|\mathbf{x}_i, a_i, o_i) = p(\cdot|\mathbf{x}_i, a_i, o_i)$ almost surely under the training distribution.

Then ω_* is a population minimizer of $\mathcal{L}_{\mathsf{CWM}}$, and every population minimizer recovers the true counterfactual conditional distribution almost surely under the training distribution.

Proof. By assumption, $\mathbf{u}_i$ is independent of $(\mathbf{x}_i, \hat{\mathbf{x}}_i, a_i, \hat{a}_i)$ before the two next states are generated. Conditional on the factual state and action, the factual next state depends on $\mathbf{u}_i$ through $\hat{\mathbf{x}}_{i+1} = f(\hat{\mathbf{x}}_i, \hat{a}_i, \mathbf{u}_i)$. Consequently, after conditioning on $o_i = (\hat{\mathbf{x}}_i, \hat{a}_i, \hat{\mathbf{x}}_{i+1})$, the queried state and action provide no additional information about $\mathbf{u}_i$:

$$p(\mathbf{u}_i|\mathbf{x}_i, a_i, o_i) = p(\mathbf{u}_i|o_i).$$

Since $\mathbf{x}_{i+1} = f(\mathbf{x}_i, a_i, \mathbf{u}_i)$, its conditional distribution given $(\mathbf{x}_i, a_i, o_i)$ is the distribution obtained by applying $f(\mathbf{x}_i, a_i, \cdot)$ to $\mathbf{u}_i \sim p(\cdot|o_i)$, which is the counterfactual distribution $p(\cdot|\mathbf{x}_i, a_i, o_i)$ (7).

Integrability and the second assumption allow us to subtract the expected loss at ω_* from at ω:

$$\begin{aligned}\mathcal{L}_{\mathsf{CWM}}(\omega) - \mathcal{L}_{\mathsf{CWM}}(\omega_*) &= \sum_{i=0}^{n-1} \mathbb{E}\left[\log \frac{p(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}{p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}\right] \\ &= \sum_{i=0}^{n-1} \mathbb{E}\left[\mathbb{E}\left[\log \frac{p(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}{p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)} \;\middle|\; \mathbf{x}_i, a_i, o_i\right]\right] \\ &= \sum_{i=0}^{n-1} \mathbb{E}[\mathsf{KL}(p(\cdot|\mathbf{x}_i, a_i, o_i) \,\|\, p_\omega(\cdot|\mathbf{x}_i, a_i, o_i))]\,.\end{aligned} \tag{14}$$

The last equality follows from the fact that that the conditional distribution of $\mathbf{x}_{i+1}$ conditioned on $(\mathbf{x}_i, a_i, o_i)$ is the counterfactual distribution $p(\cdot|\mathbf{x}_i, a_i, o_i)$ (7), from the definition of KL divergence,

$$\begin{aligned}\mathbb{E}\left[\log \frac{p(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}{p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)} \;\middle|\; \mathbf{x}_i, a_i, o_i\right] &= \mathbb{E}_{\mathbf{x}_{i+1}\sim p(\cdot|\mathbf{x}_i, a_i, o_i)}\left[\log \frac{p(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}{p_\omega(\mathbf{x}_{i+1}|\mathbf{x}_i, a_i, o_i)}\right] \\ &= \mathsf{KL}(p(\cdot|\mathbf{x}_i, a_i, o_i) \,\|\, p_\omega(\cdot|\mathbf{x}_i, a_i, o_i))\,.\end{aligned}$$

Every term on the right-hand side of (14) is nonnegative, so ω_* is a population minimizer. For any other population minimizer, the left-hand side is zero. Since the sum has finitely many nonnegative terms, each conditional KL divergence must be zero almost surely under the corresponding training distribution. The equality condition for KL divergence then gives

$$p_\omega(\cdot|\mathbf{x}_i, a_i, o_i) = p(\cdot|\mathbf{x}_i, a_i, o_i) \tag{15}$$

almost surely, for every i. □

B.3 Proof of Proposition 3.3

Proposition 3.3. *Assume that all noise variables in the SCM (6) are mutually independent and have distributions independent of θ. Let $m \geq 2$ and $\hat{\tau} \sim p_\theta$. At every step i, sample the counterfactual trajectories as in (10), using exact counterfactual transition distributions and fresh independent policy randomness. Under the usual regularity and integrability conditions, RLB is unbiased.*

Proof. Fix a step i. All trajectories considered below start from $\hat{\mathbf{x}}_i$, and the trajectory for $j = 1$ is the suffix of the factual trajectory.

By independence of the noises and the SCM factorization, the posterior future environmental noises are independent:

$$p(\mathbf{u}_i, \ldots, \mathbf{u}_{n-1}|\hat{\tau}) = \prod_{k=i}^{n-1} p(\mathbf{u}_k|o_k).$$

Thus each exact counterfactual trajectory can equivalently be generated by drawing the future environmental noises from this posterior and following π_θ with fresh policy randomness. By the tower property, averaging this posterior over $\hat{\tau}$ conditional on $(\hat{\mathbf{x}}_i, \hat{a}_i)$ gives $p(\mathbf{u}_i, \ldots, \mathbf{u}_{n-1}|\hat{\mathbf{x}}_i, \hat{a}_i)$. The future environmental noises are independent of $(\hat{\mathbf{x}}_i, \hat{a}_i)$, so this is their original joint distribution. Consequently,

$$\tau_{\geq i}^{(j)}|(\hat{\mathbf{x}}_i, \hat{a}_i) \sim p_\theta(\cdot|\hat{\mathbf{x}}_i), \qquad j = 2, \ldots, m.$$

In particular, each trajectory has the ordinary marginal distribution after averaging over the factual trajectory. Hence, including $j = 1$,

$$\mathbb{E}[G_i^{(j)}\mathbf{z}_i^{(j)}] = \mathbb{E}_{\tau\sim p_\theta}[G_i\mathbf{z}_i], \qquad j = 1, \ldots, m. \tag{16}$$

We now show that the factual baseline $j = 1$ contributes zero. The conditional distribution established above gives

$$\mathbb{E}[b_i^{(1)}|\hat{\mathbf{x}}_i, \hat{a}_i] = \frac{1}{m-1}\sum_{\ell=2}^{m} \mathbb{E}[G_i^{(\ell)}|\hat{\mathbf{x}}_i, \hat{a}_i] = \mathbb{E}[G_i^{(1)}|\hat{\mathbf{x}}_i].$$

The policy score has conditional mean zero, since

$$\mathbb{E}[\mathbf{z}_i^{(1)}|\hat{\mathbf{x}}_i] = \sum_{a\in A} \pi_\theta(a|\hat{\mathbf{x}}_i)\nabla_\theta \log \pi_\theta(a|\hat{\mathbf{x}}_i) = \nabla_\theta \sum_{a\in A} \pi_\theta(a|\hat{\mathbf{x}}_i) = 0.$$

Because $\mathbf{z}_i^{(1)}$ is determined by $(\hat{\mathbf{x}}_i, \hat{a}_i)$, conditioning first on this pair yields

$$\mathbb{E}[b_i^{(1)}\mathbf{z}_i^{(1)}|\hat{\mathbf{x}}_i] = \mathbb{E}[G_i^{(1)}|\hat{\mathbf{x}}_i]\,\mathbb{E}[\mathbf{z}_i^{(1)}|\hat{\mathbf{x}}_i] = 0.$$

We now show that every counterfactual baseline $j \geq 2$ contributes zero. For $j \geq 2$, conditional on $\hat{\tau}$, the factual return is fixed and all other counterfactual trajectories are independent of trajectory j. Thus $b_i^{(j)}$ and $\mathbf{z}_i^{(j)}$ are conditionally independent. Moreover, $a_i^{(j)}$ is drawn from $\pi_\theta(\cdot|\hat{\mathbf{x}}_i)$ even conditional on $\hat{\tau}$, so the same score identity gives $\mathbb{E}[\mathbf{z}_i^{(j)}|\hat{\tau}] = 0$. Therefore,

$$\mathbb{E}[b_i^{(j)}\mathbf{z}_i^{(j)}|\hat{\tau}] = \mathbb{E}[b_i^{(j)}|\hat{\tau}]\,\mathbb{E}[\mathbf{z}_i^{(j)}|\hat{\tau}] = 0.$$

Together, we have $\mathbb{E}[b_i^{(j)}\mathbf{z}_i^{(j)}] = 0$ for every i, j. Using (16) and the policy gradient identity (2),

$$\begin{aligned}\mathbb{E}\left[\frac{1}{m}\sum_{j=1}^{m}\sum_{i=0}^{n-1}\gamma^i(G_i^{(j)} - b_i^{(j)})\mathbf{z}_i^{(j)}\right] &= \frac{1}{m}\sum_{j=1}^{m}\sum_{i=0}^{n-1}\gamma^i\mathbb{E}_{\tau\sim p_\theta}[G_i\mathbf{z}_i] \\ &= \nabla_\theta J(\theta).\end{aligned}$$

□

B.4 Proof of Proposition 3.4

Proposition 3.4. *Consider an MDP with a fixed initial state $\mathbf{x}_0$, $n = 2$, $\gamma = 1$, and $\mathbf{x}_1 = f(\mathbf{x}_0, a_0) + \mathbf{u}_0$. Suppose $r_0 = 0$, the action at step 1 is fixed and independent of θ, and $G_0 = r_1 = \mathbf{c}^\top\mathbf{x}_1$ for a fixed vector $\mathbf{c}$. Assume f and the noise distribution are independent of θ, $\mathbf{u}_0$ is independent of a_0, and $\mathbb{E}[\|\mathbf{u}_0\|^2] < \infty$. Assume the initial policy has differentiable, positive probabilities on the finite action set A, and write $\mathbf{z}_0 = \nabla_\theta \log \pi_\theta(a_0|\mathbf{x}_0)$. For $m \geq 2$, REINFORCE and RLOO use m independent ordinary trajectories. RLB uses one factual trajectory and $m-1$ exact counterfactuals of the stated SCM, with fresh independent policy samples. Define $\widehat{\mathbf{g}}_{\mathsf{RF}} = \frac{1}{m}\sum_{j=1}^{m} G_0^{(j)}\mathbf{z}_0^{(j)}$ and, for each of RLOO and RLB, define, using the returns from the respective sampling scheme, $\widehat{\mathbf{g}} = \frac{1}{m}\sum_{j=1}^{m}(G_0^{(j)} - b_0^{(j)})\mathbf{z}_0^{(j)}$ where $b_0^{(j)} = \frac{1}{m-1}\sum_{\ell\neq j} G_0^{(\ell)}$. Let $\sigma^2 = \mathrm{Var}(\mathbf{c}^\top\mathbf{u}_0)$ and $L = \max_{a\in A}|\mathbf{c}^\top f(\mathbf{x}_0, a) - \mathbb{E}[\mathbf{c}^\top f(\mathbf{x}_0, a_0)]|$. Then:*

- *(i) RLB reduces covariance relative to RLOO: $\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLOO}}) - \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) = \frac{\sigma^2}{m-1}\mathbb{E}[\mathbf{z}_0\mathbf{z}_0^\top] \succeq 0$. The trace reduction is strict if $\sigma^2 > 0$ and $\mathbb{E}[\|\mathbf{z}_0\|^2] > 0$.*
- *(ii) If also $\sigma^2 \geq \frac{m}{m-1}L^2$, RLB reduces covariance over REINFORCE: $\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) \preceq \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RF}})$.*

Proof. The step-1 policy score is zero, so only step 0 contributes to the gradient estimators. Throughout, expectations are evaluated at the fixed policy parameter θ.

Step 1: Rewrite the leave-one-out estimator. Couple all three estimators using the same independent actions $a_0^{(1)}, \ldots, a_0^{(m)}$. Abbreviate

$$\mathbf{z}_j = \mathbf{z}_0^{(j)}, \qquad \bar{\mathbf{z}} = \frac{1}{m}\sum_{j=1}^{m}\mathbf{z}_j, \qquad \mathbf{S} = \mathbb{E}[\mathbf{z}_0\mathbf{z}_0^\top].$$

The policy score has mean zero:

$$\mathbb{E}[\mathbf{z}_0] = \sum_{a\in A}\pi_\theta(a|\mathbf{x}_0)\nabla_\theta\log\pi_\theta(a|\mathbf{x}_0) = \nabla_\theta\sum_{a\in A}\pi_\theta(a|\mathbf{x}_0) = 0.$$

Directly expanding the leave-one-out baseline gives

$$\frac{1}{m}\sum_{j=1}^{m}(G_0^{(j)} - b_0^{(j)})\mathbf{z}_j = \frac{1}{m-1}\sum_{j=1}^{m}G_0^{(j)}(\mathbf{z}_j - \bar{\mathbf{z}}).$$

Step 2: Cancel the environmental contribution in RLB. The factual transition determines

$$\mathbf{u}_0 = \mathbf{x}_1^{(1)} - f(\mathbf{x}_0, a_0^{(1)}).$$

Exact counterfactuals therefore share this same noise. Its contribution cancels because $\sum_j (\mathbf{z}_j - \bar{\mathbf{z}}) = 0$. Define

$$h(a) = \mathbf{c}^\top f(\mathbf{x}_0, a) - \mathbb{E}[\mathbf{c}^\top f(\mathbf{x}_0, a_0)], \qquad h_j = h(a_0^{(j)}).$$

The constant subtracted in h also cancels, giving

$$\widehat{\mathbf{g}}_{\mathsf{RLB}} = \frac{1}{m-1} \sum_{j=1}^{m} h_j (\mathbf{z}_j - \bar{\mathbf{z}}).$$

In particular, this estimator depends only on the sampled actions, and $\mathbb{E}[h(a_0)] = 0$ and $|h(a)| \le L$.

Step 3: Compare RLB with RLOO. Conditional on the sampled actions, RLOO uses independent environmental noises. Its conditional mean equals $\widehat{\mathbf{g}}_{\mathsf{RLB}}$, since the common mean noise contribution cancels. Its conditional covariance is

$$\frac{\sigma^2}{(m-1)^2} \sum_{j=1}^{m} (\mathbf{z}_j - \bar{\mathbf{z}})(\mathbf{z}_j - \bar{\mathbf{z}})^\top.$$

The law of total covariance consequently gives

$$\begin{aligned} &\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLOO}}) - \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) \\ &\quad = \frac{\sigma^2}{(m-1)^2} \mathbb{E}\left[\sum_{j=1}^{m} (\mathbf{z}_j - \bar{\mathbf{z}})(\mathbf{z}_j - \bar{\mathbf{z}})^\top \right]. \end{aligned}$$

Independence and the zero mean of the scores imply

$$\mathbb{E}[\bar{\mathbf{z}}\bar{\mathbf{z}}^\top] = \frac{1}{m}\mathbf{S}.$$

Therefore the expectation of the sum above is $m\mathbf{S} - m(\mathbf{S}/m) = (m-1)\mathbf{S}$. This proves part (i).

Step 4: Bound the remaining covariance of RLB. Write

$$\mathbf{H} = \frac{1}{m} \sum_{j=1}^{m} h_j \mathbf{z}_j, \qquad \mathbf{K} = \frac{1}{m(m-1)} \sum_{j \neq k} h_j \mathbf{z}_k.$$

Expanding the expression in Step 2 shows that $\widehat{\mathbf{g}}_{\mathsf{RLB}} = \mathbf{H} - \mathbf{K}$. For this calculation, set

$$v = \mathbb{E}[h(a_0)^2], \qquad \mathbf{d} = \mathbb{E}[h(a_0)\mathbf{z}_0].$$

Independence across samples gives

$$\mathbb{E}[\mathbf{H}] = \mathbf{d}, \qquad \mathbb{E}[\mathbf{K}] = 0,$$

and

$$\mathrm{Cov}(\mathbf{H}) = \frac{1}{m}\left(\mathbb{E}[h(a_0)^2 \mathbf{z}_0 \mathbf{z}_0^\top] - \mathbf{d}\mathbf{d}^\top \right).$$

Also, $\mathrm{Cov}(\mathbf{H}, \mathbf{K}) = 0$. Indeed, every term in $\mathbb{E}[\mathbf{H}\mathbf{K}^\top]$ contains either an independent score with mean zero or an independent h_j with mean zero.

To compute $\mathrm{Cov}(\mathbf{K})$, expand its second moment. A term indexed by ordered pairs (j,k) and (p,q) can be nonzero only when the pairs are identical or reversed. Identical pairs contribute $v\mathbf{S}$, and reversed pairs contribute $\mathbf{d}\mathbf{d}^\top$. There are $m(m-1)$ pairs of each type, so

$$\mathrm{Cov}(\mathbf{K}) = \frac{v\mathbf{S} + \mathbf{d}\mathbf{d}^\top}{m(m-1)}.$$

Combining these identities yields

$$\begin{aligned} \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) = \frac{1}{m}\mathbb{E}[h(a_0)^2 \mathbf{z}_0 \mathbf{z}_0^\top] + \frac{v}{m(m-1)}\mathbf{S} \\ - \frac{m-2}{m(m-1)}\mathbf{d}\mathbf{d}^\top. \end{aligned}$$

Since $m \geq 2$, $h(a)^2 \leq L^2$, and $v \leq L^2$,

$$\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}) \preceq \left(\frac{L^2}{m} + \frac{L^2}{m(m-1)}\right)\mathbf{S} = \frac{L^2}{m-1}\mathbf{S}.$$

Step 5: Compare RLB with REINFORCE. Conditional on a_0, the score is fixed and the return has variance σ^2. Hence

$$\mathrm{Cov}(G_0\mathbf{z}_0 \mid a_0) = \sigma^2\mathbf{z}_0\mathbf{z}_0^\top.$$

REINFORCE averages m independent trajectories. The law of total covariance therefore gives

$$\begin{aligned}\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RF}}) &= \frac{\sigma^2}{m}\mathbf{S} + \frac{1}{m}\,\mathrm{Cov}(\mathbb{E}[G_0 \mid a_0]\mathbf{z}_0) \\ &\succeq \frac{\sigma^2}{m}\mathbf{S}.\end{aligned}$$

Under the additional assumption in part (ii),

$$\mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RF}}) \succeq \frac{L^2}{m-1}\mathbf{S} \succeq \mathrm{Cov}(\widehat{\mathbf{g}}_{\mathsf{RLB}}).$$

This proves part (ii). □

C Experiment details

C.1 Contextual bandit

Environment and agents. We adapt the setting of Mesnard et al. (2020), using deterministic reward as outlined in Section 5.1. The policy is a linear layer followed by softmax to give 21 action probabilities, and it receives the context $\mathbf{c} \in \{-10, ..., 10\}$ which is represented by a 21-dimensional one-hot vector. For REINFORCE with baseline, we use a separate linear function of the context, fitted by squared loss. For CCA, we follow Mesnard et al. (2020) to reproduce the method and apply it to our setting. All methods use Adam with learning rate 4×10^{-4} and entropy regularization of 0.004. Each training uses 10^7 environmental interactions, with batch size 256 per update.

Offline model learning. For world model and counterfactual world model training, we use REINFORCE with baseline as the reference policy, with its action probabilities equally mixed with a uniform policy. We create paired data using uniformly sampled contexts and two independent actions, and the same environmental noise. We collect 4,096 training, 1,024 validation, and 2,048 test pairs, keeping both members of every pair in the same split. This costs 14,336 environmental interactions per dataset.

Within next state $\mathbf{x} = (\mathbf{c}, a + \mathbf{u})$, both models are trained to predict $a + \mathbf{u}$ since $\mathbf{c}$ is already provided as input. The models use the same paired dataset, where the ordinary world model treats both pair members as separate examples conditioned on context and action. The counterfactual model trains on both directions, conditioned on one and predicting the other, and the other way around as well.

Each model has two hidden layers of eight SiLU units, Gaussian mean and scale outputs, and a learned linear connection from inputs to the mean. This makes these models slightly smaller than the policy. For both models, we clip the predicted Gaussian standard deviation to have a floor of 0.05, so that maximum likelihood does not explode for a constant prediction target.

Each model is trained with 5,000 Adam updates with 256 examples sampled with replacement per update. We use cosine learning rate scheduler from 3×10^{-3} to 3×10^{-5}. After training, we use the final checkpoint and keep it frozen throughout policy learning.

C.2 Multi-agent bandit

Environment and agents. Based on the contextual bandit problem, we adapt the multi-agent setting of Mesnard et al. (2020). Contexts are independent across agents, and actions are sampled independently given the contexts and current policies. Each agent observes only its own context before acting. Each agent has its own policy and baseline with the architecture and optimization settings of Appendix C.1, and parameters are not shared between agents.

Offline model learning. Similarly as in contextual bandit, we use the agents trained with REINFORCE with baseline as the reference policy, with each agent's policy mixed equally with uniform actions. We generate data in groups, where each group contains a factual joint episode across the agents and one alternative for each agent with shared environmental randomness (all other agents). Herein, each alternative resamples only that agent's action while retaining all contexts and other agents' factual actions. This gives one pair per agent using $K + 1$ joint environment interactions in total for all agents. We collect 4,096 training, 1,024 validation, and 2,048 test groups per seed, keeping every observation from a group in the same split.

Each agent has its own ordinary and counterfactual world model which predicts the scalar terminal cost $\mathbf{x}^i = -r$. The ordinary model takes $(\mathbf{c}^i, a^i)$, and the counterfactual model additionally takes the factual observation o^i defined in Section 5.2. Neither observes other agents' contexts, actions, or individual costs. Both use two hidden layers of 16 SiLU units and a learned linear connection to the mean, again yielding models slightly smaller than the policy.

The models use Gaussian densities restricted to $[0, 400K]$, the possible joint-cost interval. These are continuous approximations to integer costs and the underlying Gaussian scale has a floor of 0.05 to prevent explosion. Like as in contextual bandits, ordinary world models train on both pair elements and counterfactual models are trained on both directions. Each independent model receives 5,000 updates with 256 examples per update and the same Adam schedule as in Appendix C.1.

C.3 The dinosaur game

C.3.1 Environment and observations

Chrome Dino is an endless side-scrolling game. The dinosaur remains at a fixed horizontal location while obstacles move from right to left. We use three actions: stand, jump, and duck. The policy observes the four most recent decision frames in grayscale at 128×64. One decision repeats the selected action for four raw game frames.

In the distributed game, jumping is an almost universal solution. We modify the original game so that the three actions have distinct consequences. Cacti and low birds require a jump, middle birds require a duck, and high birds should be ignored. We increase the frequency of birds and allow them to appear throughout an episode. The same modified simulator is used to collect world-model data, construct direct-training roots, and evaluate every policy.

The reward is $+1$ when an obstacle is passed, -1 upon collision, and zero otherwise. Training rollouts continue after a collision and have a fixed length of $H = 20$ decisions, or 80 raw frames. Evaluation terminates at the first collision or after 375 decisions.

A branch root is a state from which multiple policy rollouts are generated. We branch when the nearest obstacle satisfies $8 < \mathrm{TTC} \leq 12$ raw frames, where

$$\mathrm{TTC} = \frac{\text{horizontal distance to the nearest obstacle}}{\text{game speed}}.$$

Roots are sampled equally from four speed ranges spanning the game. Direct roots are constructed with stand and duck poses in proportions matched to the video-root pool. Video roots are extracted from recorded episodes, and roots following a recent jump are excluded to avoid branching during an unresolved jump.

C.3.2 Diffusion World Model

The diffusion model is trained on $1{,}000$ recorded episodes of 150 decisions each under the modified game, for $150{,}000$ observed decisions. Each frame is an RGB image with 320×240 resolution. Training examples contain eight preceding frames, the single causal action, and the next frame. A pretrained VAE maps frames to latent space. We jointly train an action embedding and a 101M-parameter conditional U-Net using the velocity-prediction objective. Training uses AdamW with learning rate 2×10^{-4}, gradient clipping at norm 1, batch of 32 across eight devices, bfloat16 computation, EMA with warmup and decay 0.9999. We apply Classifier-Free Guidance (Ho & Salimans (2022)) by dropping the action with probability 0.2 during training. All experiments use the EMA checkpoint after $9{,}000$ steps.

We generate each frame with 10 deterministic DDIM steps with guidance scale 1.0. The generated frame is appended to the context, so a policy rollout is generated autoregressively for $H = 20$ decisions. A frozen visual classifier maps each pair of consecutive generated frames to collision, no-event, or obstacle-clear reward. For policy input, generated frames are converted to grayscale and resized to 128×64.

For RLB, at every rollout step, DDIM inversion uses the corresponding factual eight-frame context, factual action, and factual next frame to abduct factual noise. This noise is shared across the four group members, while each member conditions on its sampled action and recursively generated context. RLOO and both REINFORCE methods instead use independent fresh Gaussian noise.

C.3.3 Policy optimization

All methods use the same convolutional policy. Its three convolutional layers have 32, 64, and 64 channels, kernel sizes 8, 4, and 3, and strides 4, 2, and 1. We use orthogonal initialization, Adam with learning rate 10^{-3}, gradient clipping at norm 20, discount $\gamma = 0.99$, and entropy coefficient 0.04. The learned-baseline control uses a separate value network with the same convolutional architecture and a scalar output, optimized with learning rate 3×10^{-2}.

Each update contains 32 trajectories. RLB and RLOO arrange these as eight groups of $m = 4$ trajectories. Every method is trained for $2{,}000$ updates and three seeds, giving $64{,}000$ trajectories and 1.28 million rollout decisions per seed.

C.3.4 Validation and testing

We save a checkpoint every 100 policy updates. For each method, every checkpoint is evaluated on 30 validation episodes for each training seed. We select one checkpoint index using its mean validation performance across the three seeds. The selected checkpoint is then evaluated once on 100 test episodes per seed. Validation and test episodes use fixed and different simulator seeds that are shared by all methods.

All policies predict action greedily in the real simulator. An episode ends at the first collision or after 375 decisions. We report the fraction of encountered obstacles that cause a collision and the mean number of decisions survived before the first collision. Survival is capped at 375 decisions. The main table reports the mean and sample standard deviation across the three training seeds.

C.3.5 Counterfactual-quality protocol

We follow the four criteria of Melistas et al. (2024). For each root we retain the factual action and uniformly select one of the other two actions as the intervention. CWM and WM therefore receive exactly the same contexts and intervention actions. CWM uses the DDIM-inverted factual noise and WM uses fresh Gaussian noise.

Composition is the mean pixel ℓ_1 error, on the 0–255 scale, between the recorded next frame and a generation made under the factual action. Effectiveness is macro-F1 from a trained action predictor applied to the generated counterfactual. Realism is FID against the recorded next frames. Minimality is counterfactual latent divergence (CLD), computed with an independently trained unconditional VAE.

We additionally use a Dino-specific minimality measure for newly entering obstacles. Existing obstacles are visible in the context frames, whereas a newly entering obstacle is not and must be generated by the model. Using only recorded frames, we select transitions where a new obstacle appears in the fixed right-edge region marked by the red dashed boxes. The obstacle-preservation score is the mean pixel ℓ_1 error between the generated counterfactual and the recorded next frame within this region, measured on the 0–255 scale.

For completeness, CLD restricted to the same preselected entering-obstacle cases is 0.310 for CWM and 0.390 for WM. The more direct event-specific pixel metric is 2.34 for CWM and 9.57 for WM, showing that CWM better preserves the factual obstacle's appearance and position.

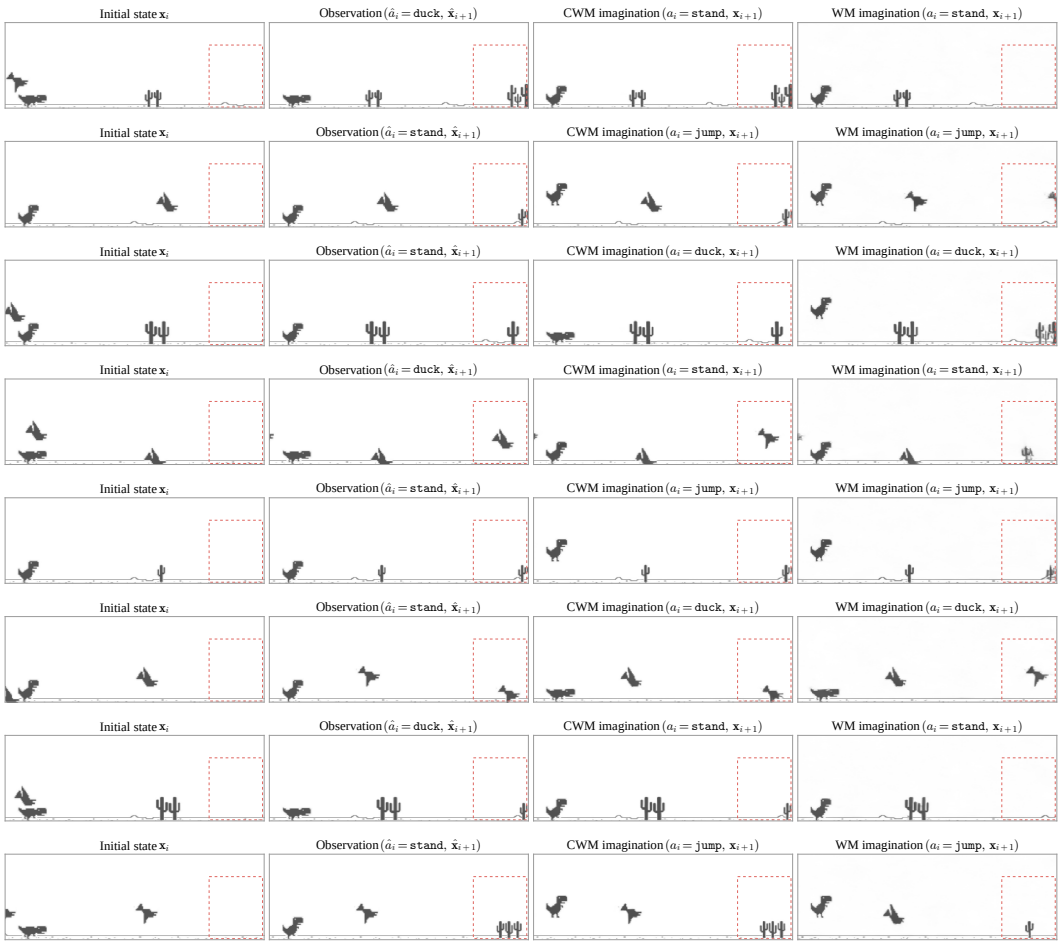

Figure 8: Eight additional examples. From left to right: initial state, recorded factual next frame, CWM generation, and ordinary WM generation. CWM and WM use the same intervention action. The red dashed boxes mark the entering-obstacle region.

C.3.6 Additional qualitative examples

Figure 8 presents eight additional comparisons using different factual and intervention actions.