When to Trust Counterfactuals: Reliable Reinforcement Learning with Partially Learned Causal Models

Anonymous authors
Paper under double-blind review

Abstract

Causal reinforcement learning can improve sample efficiency by reusing factual trajectories to learn from alternative counterfactual outcomes. However, this benefit depends on the reliability of the structural causal model (SCM) used to generate counterfactual trajectories. We study this problem through Counterfactually Guided Proximal Policy Optimization (CG-PPO) with a partially learned SCM, where known mechanisms remain explicit and the unknown mechanism is represented by a model of the other agent (MOA) trained with response-equivalence supervision. We find that CG-PPO can sustain the benefit of counterfactual learning when the SCM is reliable, whereas this benefit diminishes as training proceeds when counterfactual rollouts rely on an imperfect learned SCM. We show that factual response correctness strongly predicts behavioral correctness after counterfactual branching, separating high and low reliability states by 56.4 percentage points after one step and 22.4 points after 40 steps. This discrimination remains positive across the observed range of factual–counterfactual trajectory divergence. We use this signal to introduce reliability-aware branch-state sampling through Consecutive Correctness Weighting (CCW) and Censor-Aware Percentile Weighting (CAPW). When counterfactual PPO updates are interleaved with factual PPO updates, CAPW raises mean win rate over the final six evaluations from 57.0% to 63.0% relative to matched uniform sampling, with higher Final MA6 in all five matched training seeds. It also achieves a 13.7% relative improvement in normalized learning-curve area with respect to factual environment exposure over factual-only PPO, with improvements across all five matched seeds. Under stochastic dynamics, factual conditioning additionally improves learning under the tested SCM misspecification but provides no consistent advantage when the SCM correctly models the exogenous-noise distribution. These results show that locally measured reliability can guide counterfactual training-data allocation when reinforcement learning relies on partially learned causal models.

1 Introduction

Reinforcement learning often requires substantial environment interaction, whereas counterfactual reasoning can extract additional policy-learning information from factual experience by asking how alternative actions or policies would have unfolded under the same observed conditions (Buesing et al., 2019). Structural causal models (SCMs) provide a formal basis for such interventions and counterfactuals (Pearl, 2009; Peters et al., 2017). Causal reinforcement learning methods differ, however, in how much of the required causal information is assumed known and how much must be learned (Zeng et al., 2024; Deng et al., 2023). We study an intermediate setting in which the causal graph and most mechanisms are specified, but one behaviourally important mechanism must be learned. The central question is whether factual evidence about that mechanism's reliability can guide where counterfactual training data should be generated.

We study this question in a deceptive invader–defender environment in which the invader has a private true goal and can alter its motion towards other candidate goals before redirecting. The defender infers the invader's intended goal from its motion and responds accordingly, creating a causal chain from invader action, through inferred intent, to defender response. Related work has studied motion

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.

and control designed to influence an observer's inferred goal (Dragan et al., 2015; Ornik & Topcu, 2018), while opponent-modelling methods have similarly estimated latent goals from observed behaviour (Raileanu et al., 2018). In our setting, the physical mechanisms and defender response rule are specified, while a Model of the Other Agent (MOA) learns the defender's latent believed goal.

Our framework, Counterfactually Guided Proximal Policy Optimization (CG-PPO), follows the factual grounding principle of Counterfactually-Guided Policy Search (CF-GPS) (Buesing et al., 2019), while using Proximal Policy Optimization (PPO) (Schulman et al., 2017) for policy updates. The MOA is trained with response-equivalence supervision, treating latent goal labels as equivalent when they induce the same defender response, and is used only for counterfactual rollout. Our central finding is that reliability observed on factual experience remains informative after counterfactual intervention. Factual persistence of MOA response correctness strongly ranks its correctness after branching across the range of factual–counterfactual trajectory divergence observed under CG-PPO. We use this signal to allocate counterfactual training toward more reliable branch states through Consecutive Correctness Weighting (CCW) and a censor-aware variant, CAPW. In mixed factual–counterfactual learning, CAPW improves learning efficiency relative to factual-only PPO and improves final performance relative to uniform counterfactual sampling. A separate stochastic study examines factual reuse of exogenous variables under SCM misspecification. A controlled comparison shows that factual PPO shifts the relative selector effect toward CAPW, consistent with the broader branch-state support from its censor-aware weighting. Separately, factual exogenous reuse improves learning under the tested SCM misspecification but offers no consistent advantage under the accurate noise model.

2 Related work

Counterfactual reinforcement learning uses factual experience to reason about actions or policies that were not executed. Counterfactually-Guided Policy Search (CF-GPS) grounds such rollouts in factual trajectories through a structural causal model (SCM) (Buesing et al., 2019). Counterfactual semantics, however, depend on assumptions beyond a fitted transition distribution. Gumbel-Max SCMs provide one construction for sequential counterfactual evaluation (Oberst & Sontag, 2019), while generalized causal mechanisms show that models compatible with observed behaviour need not imply the same counterfactuals (Lorberbom et al., 2021). Our setting instead assumes that the causal graph and most mechanisms are specified, while one mechanism required for counterfactual rollout must be learned.

Learned-model reinforcement learning similarly faces the problem of deciding when synthetic experience is sufficiently reliable for policy learning. Model-Based Policy Optimization limits compounding model error through short rollouts initialized from real states (Janner et al., 2019). Masked Model-based Actor-Critic restricts learned-model use according to local state-action uncertainty (Pan et al., 2020). MACURA, introduced by Frauenknecht et al. (2024), similarly adapts model rollout usage based on local uncertainty. These approaches show that learned-model use can be controlled locally rather than accepted uniformly. Our setting differs in that reliability is measured from the observed behavioural correctness of one learned causal mechanism along factual trajectories and is then evaluated after genuine counterfactual branching.

Opponent modelling provides a related basis for learning behaviourally important latent quantities. Self Other-Modeling estimates another agent's hidden goal from observed behaviour (Raileanu et al., 2018), while later work infers opponent subgoals from historical trajectories (Yu et al., 2024). Our MOA similarly represents a latent goal-related quantity, but it is used as a causal mechanism for counterfactual rollout rather than as an input to the invader policy. Its supervision also targets behavioural equivalence rather than unique latent-label recovery. This is related to value-equivalent model learning, which relaxes exact model fidelity by preserving only the distinctions needed for planning (Grimm et al., 2020), and to partial-label learning, where multiple labels may be compatible with a training example (Cour et al., 2011). Our response-equivalence criterion is narrower, since latent goals are treated as equivalent only when they induce the same defender response.

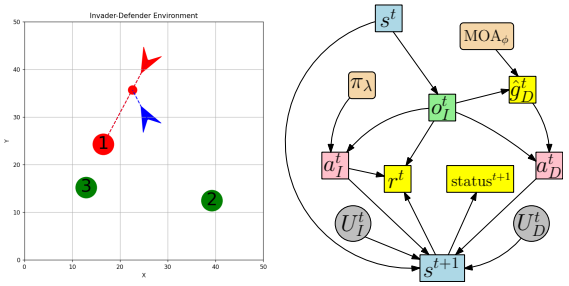
3 INVADER–DEFENDER PROBLEM FORMULATION

We consider an episodic invader–defender task with candidate-goal set $\mathcal{G} = \{g_0, g_1, g_2\}$ in the experiments. The invader has a private true goal $g^* \in \mathcal{G}$, while the defender does not observe g^* and must infer the invader's intended goal from motion. The invader wins by reaching g^* before interception. The defender wins by intercepting the invader first. Each episode is generated so that the defender is initially closer than the invader to every candidate goal, giving the defender a geometric advantage. The invader partly offsets this with a maximum steering limit of $\pi/8$, which is twice that of the defender, but successful policies must still exploit defender uncertainty rather than simply race directly to the true goal. Figure 1(a) shows the environment geometry.

The invader sees a partial observation and chooses among three discrete actions, each directing it toward a candidate goal. The defender's believed invader goal is hidden from the policy. Conditional on that belief, its response and the physical dynamics are specified. Episodes end at goal arrival, interception, or 200 steps. Training uses terminal rewards and bounded influence shaping, while evaluation measures goal-reaching win rate. The policy maximizes expected discounted environment return with discount $\gamma = 0.99$ and is evaluated in the environment. Appendix A gives the objective, dynamics, observations, capture calculation, and reward specification.

4 PARTIALLY LEARNED STRUCTURAL CAUSAL MODEL

We model factual and counterfactual rollout with a partially learned structural causal model (SCM). Figure 1(b) shows the one-step SCM. The graph structure, defender response rule, physical transition, reward, observation, and termination mechanisms are specified from the environment. The only learned causal mechanism is the Model of the Other Agent (MOA), parameterized by ϕ, which predicts the defender's latent believed goal. The defender response conditional on that belief remains a known mechanism.



(a) Environment visual (b) Partially learned SCM

Figure 1: Invader–defender environment and partially learned structural causal model.

With hats denoting generated quantities, a counterfactual step is

$$\hat{g}_{D,t} = m_\phi(\hat{x}_t) = \arg\max_{g \in \mathcal{G}} q_\phi(g \mid \hat{x}_t), \tag{1}$$

$$\hat{a}_{D,t} = f_D(\hat{s}_t, \hat{g}_{D,t}), \qquad \hat{s}_{t+1} = f_T(\hat{s}_t, \hat{a}_t, \hat{a}_{D,t}, u_t^I, u_t^D), \tag{2}$$

$$\hat{r}_t = f_R(\hat{s}_t, \hat{a}_t, \hat{a}_{D,t}, \hat{s}_{t+1}). \tag{3}$$

Here q_ϕ is the categorical MOA output, $\hat{x}_t$ contains geometry and short-history features derived from $\hat{s}_t$, and $\hat{a}_{D,t}$ is the defender response. Observation and termination are also known structural functions. At a selected factual branch index, CG-PPO retains the factual state and available exogenous suffix, replaces the invader's factual action mechanism with a planner policy, and rolls the partially learned SCM forward. The MOA is re-evaluated on generated states after branching. The defender belief is not fixed to the factual latent label.

The graph also includes exogenous heading disturbances U_t^I and U_t^D for the invader and defender. Both are parents of s_{t+1} and do not directly affect the MOA, defender belief, or defender action. In the deterministic core reliability experiments, these disturbances are zero. In the stochastic study, they are either reused from the matched factual suffix or sampled from the SCM law, depending on treatment. Their presence in Figure 1(b) therefore gives a general SCM for both settings without implying that the main reliability experiments were stochastic.

5 COUNTERFACTUALLY GUIDED PROXIMAL POLICY OPTIMIZATION

CG-PPO retains the factual-grounded rollout principle of CF-GPS (Buesing et al., 2019) but uses PPO (Schulman et al., 2017) for policy improvement. Figure 2 summarizes the framework. An *update unit* is a batch of valid timesteps for one PPO update. Factual units come from environment trajectories. Counterfactual units branch from retained factual states and roll the SCM forward under a planner policy. Both train the same target policy.

Factual trajectories provide PPO data, MOA supervision, and candidate branch states. The allocator sets CF exposure and selects branch states, while the planner supplies alternative actions for SCM rollout. Our selectors change branch allocation without changing rollout or PPO optimization.

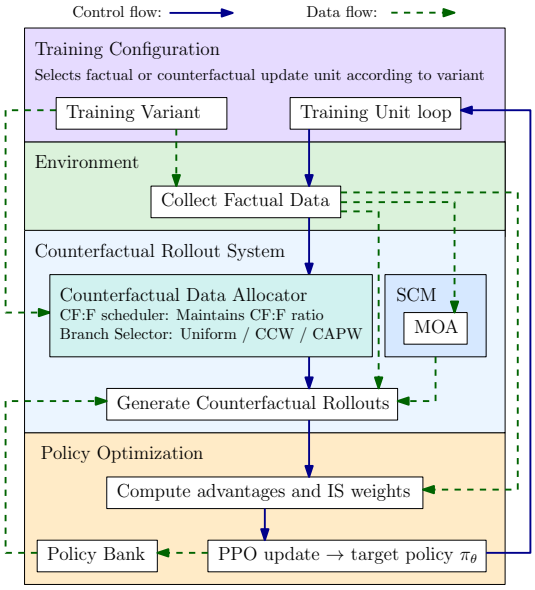


Figure 2: CG-PPO with a partially learned SCM and reliability-aware branch allocation. IS denotes importance sampling.

5.1 MODEL OF THE OTHER AGENT AND RESPONSE-EQUIVALENCE SUPERVISION

We train the categorical MOA output $q_\phi(g \mid x)$ using response-equivalence supervision. Its features x include current state and short motion history. The MOA supplies the latent defender belief during counterfactual rollout.

The latent believed goal need not be uniquely identifiable from the defender's observed response. Let $\mathrm{resp}(s, g)$ denote the defender response produced by the specified response rule at state s under believed goal g, and let $\mathrm{resp}_{\mathrm{fact}}$ denote the corresponding factual response. We define the response-equivalent set

$$\mathcal{G}_{\mathrm{eq}}(s) = \{g \in \mathcal{G} : \mathrm{resp}(s, g) \sim \mathrm{resp}_{\mathrm{fact}}\}. \tag{4}$$

Here $\sim$ means that responses match under the specified tolerance. MOA training uses response-equivalence supervision rather than requiring a unique latent label. Its loss is

$$\mathcal{L}_{\mathrm{MOA}} = -\log\left(\sum_{g \in \mathcal{G}_{\mathrm{eq}}(s)} q_\phi(g \mid x)\right) + \sum_{g \in \mathcal{G}} q_\phi(g \mid x)\, \Delta(s, g). \tag{5}$$

where $\Delta(s, g)$ measures the discrepancy between the candidate and factual defender responses. The first term rewards response-equivalent predictions, while the second provides a graded penalty for the magnitude of behavioural error. The exact response representation, matching tolerance, numerical safeguards, MOA features, and optimization settings are given in Appendix B.

5.2 PLANNER, REFERENCE AND TARGET POLICIES

The *target policy* π_θ is optimized using PPO. A fixed *reference policy* π_{ref} defines its update. A recent *planner policy* π_λ from the policy bank generates counterfactual actions. Generated samples use a clipped per-step reference/planner importance ratio, while factual actions have weight one. Restricting planner–reference mismatch controls the influence of this correction. Appendix C gives the formula and implementation.

6 RELIABILITY-AWARE BRANCH-STATE SAMPLING

In our partially learned SCM, MOA errors can make counterfactual rollouts unreliable. Since these errors vary across the state space, a branch-indexed factual reliability signal can help select branch

states from which more reliable counterfactual training data can be generated. For a factual state at time t, let ℓ_t denote the number of consecutive steps, starting at t, for which the MOA predicts the defender response correctly. We use this persistence of factual correctness to select counterfactual branch states. Because we compute it retrospectively from the retained factual suffix after source collection, this statistic is specific to a branch state and does not provide a calibrated uncertainty estimate in advance.

If the factual trajectory ends before an incorrect MOA response is observed, ℓ_t is right censored because correctness may continue beyond the observed suffix. We compare three selectors that assign each eligible branch state a sampling weight ω_t.

Consecutive Correctness Weighting (CCW) favours states with longer observed correctness. It defines

$$R_{L_{\max}}(t) = \frac{\min(\ell_t, L_{\max})}{L_{\max}}, \qquad \omega_t = \epsilon + R_{L_{\max}}(t), \tag{6}$$

where $\epsilon > 0$ preserves support for states with low observed reliability. We use $\epsilon = 0.05$ and set $L_{\max} = 40$ as it approximately matches the mean counterfactual rollout length and prevents very long factual suffixes from receiving continually increasing priority.

However, for right-censored states, the observed correctness duration is only a lower bound on how long the MOA would have remained correct. When this duration is below $L_{\max}$, CCW can therefore assign a lower weight simply because the factual trajectory ended before continued correctness could be observed. This can bias intervention selection away from states with short remaining factual suffixes even when no MOA error has been observed. The limitation is expected to become more consequential as the MOA becomes more accurate because fewer correctness streaks terminate in observed mismatches and more can reach the end of the factual trajectory while still correct.

Censor-Aware Percentile Weighting (CAPW) addresses this limitation by converting the uncapped correctness duration into a reliability percentile within the current factual source while accounting for right censoring. It first constructs a weighted product-limit survival estimate using the weighted event-to-risk form of Xie & Liu (2005, Eq. (1)):

$$\hat{S}(d) = \prod_{k=1}^{d} \left(1 - \frac{E_k}{Y_k}\right), \qquad \hat{S}(0) = 1, \tag{7}$$

where $\hat{S}(d)$ estimates the weighted proportion of eligible branch states for which MOA correctness persists for at least d responses. Here, Y_k represents the trajectory-balanced risk mass observed at response depth k, while E_k represents the mass whose first mismatch occurs at that depth. We additionally weight each candidate in trajectory j by $1/|\mathcal{C}_j|$ so that every factual trajectory contributes equal total mass, analogous to the reciprocal-stratum weighting of Galimberti et al. (2002, Sec. 3.2, p. 3852):

$$\begin{aligned} Y_k &= \sum_j \frac{1}{|\mathcal{C}_j|} \left(\sum_{t \in \mathcal{U}_j} \mathbf{1}[\ell_{j,t} \geq k-1] + \sum_{t \in \mathcal{R}_j} \mathbf{1}[\ell_{j,t} \geq k] \right), \\ E_k &= \sum_j \frac{1}{|\mathcal{C}_j|} \sum_{t \in \mathcal{U}_j} \mathbf{1}[\ell_{j,t} = k-1]. \end{aligned} \tag{8}$$

Thus, Y_k is the trajectory-balanced mass observed at response depth k, while E_k is the mass whose first mismatch occurs at that depth.

After fitting $\hat{S}$, each candidate state s_t is assigned a percentile interval I_t:

$$I_t = \begin{cases} \left[1 - \hat{S}(\ell_t),\ 1 - \hat{S}(\ell_t + 1)\right], & \text{if a mismatch is observed,} \\ \left[1 - \hat{S}(\ell_t),\ 1\right], & \text{if right censored,} \end{cases} \qquad \omega_t = \mathrm{mid}(I_t). \tag{9}$$

Here, $1 - \hat{S}(\ell_t)$ gives the lower percentile bound associated with the observed correctness duration ℓ_t, namely the estimated fraction of the source-balanced reliability distribution with shorter correctness persistence. For an observed mismatch, I_t is the percentile interval associated with that exact correctness duration. For a censored state, ℓ_t provides only a lower bound, so its possible percentile

extends through the remaining upper tail. CAPW uses the midpoint of this interval as its branch-state weight rather than treating the end of the factual trajectory as an MOA mismatch.

As a baseline, Uniform sampling assigns $\omega_t = 1$ to every eligible branch state. For all three selectors, factual trajectories are sampled with equal probability, after which a branch state is sampled within the selected trajectory with probability proportional to ω_t. Thus, CCW and CAPW redistribute branch probability within trajectories rather than flattening all candidate states into a single global distribution.

7 Experimental Setup

All experiments use the fixed-defender environment described in Section 3. The core reliability and selector experiments use deterministic dynamics, while the stochastic study activates the heading disturbances in the SCM. Policies are evaluated deterministically on a configured panel of 500 held-out cases, and win rate is the fraction in which the invader reaches its true goal before capture or timeout.

To first establish the effect of counterfactual learning and SCM reliability, we also report a three-seed fixed-ratio comparison using seeds 0, 50K, and 100K over 3M factual environment steps. It includes calibrated SB3 PPO, Factual PPO using the CG-PPO policy-update implementation without counterfactual updates, learned-SCM CG-PPO using the MOA, and a reliable-SCM CG-PPO reference in which the defender-belief mechanism required for counterfactual rollout is supplied directly. Both counterfactual variants use the fixed $4{:}1$ CF:F schedule.

We distinguish the policy-update regime from the counterfactual schedule. *F* denotes factual-only PPO. *CF-only* uses factual trajectories as sources for counterfactual rollout but performs no factual PPO updates. *CFF* interleaves factual and counterfactual PPO updates. Within CFF, *R4* maintains a fixed CF:F update ratio of $4{:}1$, whereas *ID* begins at ratio 4:1 and decays immediately with factual progress. Unless otherwise stated, the main CFF selector comparison uses ID. Updates are performed in 50K-transition units. The CF-only selector comparison uses 8M optimized counterfactual transitions, while the main CFF comparison uses 3M factual and approximately 4.75M counterfactual transitions.

We summarize learning by normalized area under the win-rate learning curve (AUC) over the fixed comparison domain for each experiment. These are descriptive summaries under the reported analysis protocol. Appendix D specifies the exposure grids and normalization. Because CF-only and CFF use different training axes and budgets, AUC is used for matched comparisons within each regime rather than as a direct cross-regime score. We also report Final MA6, the mean win rate over the final six chronological evaluations. Each training seed defines one independent run, and matched treatments use the same seeds.

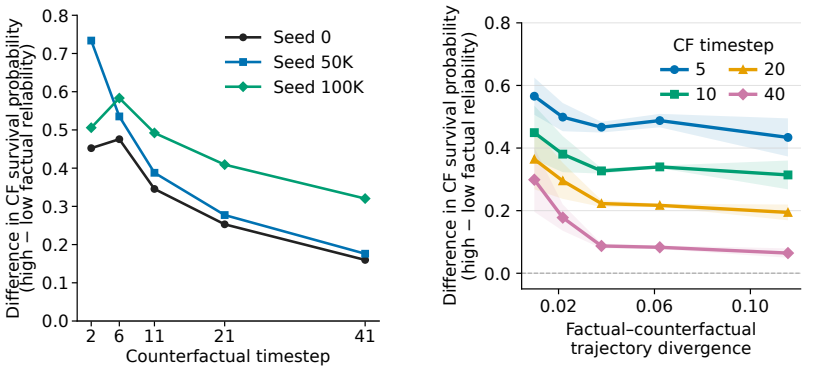
8 Results

8.1 CG-PPO with Reliable and Learned Causal Models

Figure 3 compares four variants under matched factual exposure. Both counterfactual variants learn faster than Factual PPO early. The learned-SCM variant loses this advantage as factual learning catches up, whereas the reliable-SCM reference has the highest mean full-horizon AUC and Final MA6.

The learned-SCM variant improves early policy learning: at 1M factual steps, its MA6 is 0.551 versus 0.298 for Factual PPO. Its endpoint advantage disappears, whereas the reliable-SCM reference has the highest mean AUC and Final MA6. The reference is not deployable because it directly accesses the defender-belief mechanism unavailable to the invader. Appendix F gives the full summary.

This motivates estimating where the learned mechanism is reliable before allocating counterfactual experience.



(a) Reliability transfer after branching (b) Reliability transfer across divergence

Figure 4: Factual reliability predicts counterfactual MOA correctness after branching and across realized factual–CF trajectory divergence.

8.2 Does Factual Reliability Predict Counterfactual Reliability?

The preceding comparison shows that CG-PPO's benefit depends strongly on the reliability of the causal mechanism used for counterfactual rollout. We therefore ask whether reliability measured along factual trajectories remains informative after counterfactual branching. For evaluation only, we test this using uniformly sampled branch states by comparing the MOA-induced response with the defender response generated from the simulator's true latent belief along each counterfactual rollout. Because the first counterfactual response is evaluated at the shared factual branch state, we condition on this response being correct and evaluate correctness from the following step onward.

We compare the highest and lowest within-seed factual-R_{40} bins within each factual source, then average over informative sources. Figure 4 shows the high-minus-low difference in correctness survival for each seed. Post-branch horizons 1–40 correspond to counterfactual response timesteps 2–41 because the shared first response is conditioned to be correct.

Factual reliability continues to rank MOA correctness after branching. Figure 4(a) shows a three-seed mean source-balanced high-minus-low correctness-survival gap of 0.564 after one post-branch step and 0.224 at horizon 40, with positive contrasts at every evaluated horizon in all three seeds. Figure 4(b) stratifies this comparison by realized factual–counterfactual trajectory divergence (Appendix I). The gap remains positive in every divergence quintile at CF timesteps 5, 10, 20, and 40, although long-horizon discrimination is smaller at greater realized divergence.

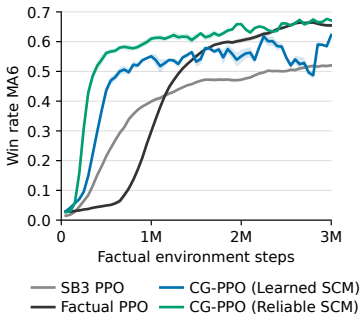
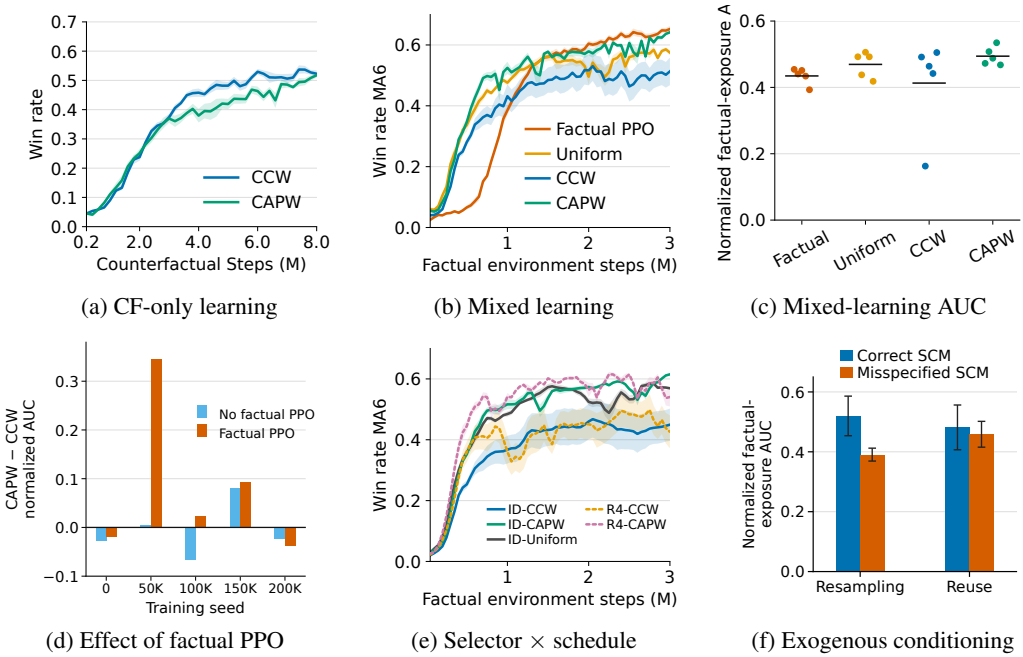


Figure 3: Fixed-defender learning under matched factual exposure. Curves show mean chronological MA6 over three seeds.

These results support factual correctness persistence as a ranking signal under the counterfactual interventions studied here. The score should be interpreted as a ranking signal rather than a calibrated probability of counterfactual correctness or a certificate of complete SCM or return accuracy. This validation also does not establish either CCW or CAPW as an optimal sampling rule.

8.3 CF-Only Selector Comparison

We first evaluate reliability-aware branch selection in isolation from factual policy updates. Under counterfactual-only policy optimization, CCW and CAPW use the same factual-source and counterfactual budgets but differ in how the factual reliability signal determines branch-state sampling. Figure 5(a) shows the resulting five-seed learning comparison.



(a) CF-only learning (b) Mixed learning (c) Mixed-learning AUC

(d) Effect of factual PPO (e) Selector × schedule (f) Exogenous conditioning

Figure 5: Selector learning and robustness analyses. Panels (a–c) compare selector learning across CF-only and mixed regimes. Panel (d) shows the CAPW–CCW interaction with factual PPO. Panel (e) compares selector performance under immediate-decay and R4 schedules. Panel (f) shows mean normalized factual-exposure AUC under stochastic resampling and factual reuse with correct and mean-misspecified SCMs; error bars are one sample SD across six seeds.

Under CF-only optimization, CCW attains the higher mean normalized AUC, 0.382 versus 0.350, while Final MA6 is closer at 0.527 versus 0.512, with heterogeneous paired effects. This differs from the mean ordering in mixed learning, but the experiments also differ in budget and schedule. This motivates examining how factual policy updates change the relative effect of the two selectors.

8.4 Reliability-Aware Counterfactual Learning

We next evaluate the selectors when counterfactual updates are interleaved with factual PPO updates under the immediate-decay CFF schedule. Figure 5(b) shows the corresponding five-seed learning curves. Under the tested immediate-decay CFF schedule, CAPW has the highest mean AUC among the selectors. Relative to matched Uniform sampling, CAPW raises normalized factual-exposure AUC from 0.469 to 0.494, with higher AUC in three of five seeds, while Final MA6 rises from 0.570 to 0.630 and final win rate from 0.575 to 0.642, with both endpoint metrics higher in all five seeds. Against factual-only PPO, CAPW improves AUC from 0.435 to 0.494 in all five seeds, although factual-only PPO attains slightly higher mean Final MA6, 0.641 versus 0.630, which is consistent with its long-run advantage from avoiding bias introduced by learned-MOA counterfactual errors. We next test the selector interaction under a matched training lifecycle.

8.5 Selector Performance Across Learning Regimes

The CF-only and mixed-CFF experiments suggest that selector performance changes with the surrounding learning regime, but those experiments also differ in training budget and update schedule. We therefore compare CAPW and CCW under the same immediate-decay lifecycle while controlling whether factual PPO updates are applied. Figure 5(d) shows the paired CAPW-minus-CCW AUC difference with and without factual PPO.

Mean CAPW-minus-CCW AUC shifts from -0.006 without factual PPO to $+0.081$ with it, toward CAPW in four of five matched seeds. Seed 50K contributes strongly to this difference, with CAPW exceeding CCW by 0.345 AUC under factual PPO. However, excluding this seed still leaves CAPW

ahead in mean AUC by 0.015 and Final MA6 by 0.051, with higher Final MA6 in all four remaining pairs. CCW also has greater across-seed variability, so seed 50K should not be treated as an isolated anomaly. The controlled comparison therefore indicates a regime-dependent shift in the relative effect of the two selectors, while leaving the downstream mechanism unresolved.

On the same candidate pools, reconstructed CAPW probabilities have broader effective support and lower selected observed correctness than CCW (Appendix H).

Figure 5(e) further shows that selector performance depends on CF schedule. ID-CCW uses 60.4% fewer optimized CF transitions than R4-CCW yet attains higher Final MA6 in all five matched seeds, by 0.126 on average. Together with the reliable-SCM comparison in Section 8.1, this is consistent with decay reducing the influence of errors propagated through the learned SCM while factual updates increasingly determine later learning. The comparison changes both CF exposure and training path and therefore does not isolate this mechanism. Under R4, CAPW also exceeds CCW in mean AUC and Final MA6, although only two of three matched seeds favour CAPW on each metric (Appendix G).

8.6 Factual Conditioning under Stochastic and Misspecified Dynamics

Counterfactual rollout under stochastic dynamics also depends on the SCM's exogenous-noise model. Resampling draws each disturbance from this model, whereas factual reuse preserves the observed disturbance where the matched factual suffix provides it and samples only beyond that support. We therefore test whether factual conditioning reduces sensitivity to exogenous-model error under an accurate noise law and a deliberately mean-misspecified one (Appendix E).

Figure 5(f) shows the resulting normalized factual-exposure AUC. With the correct SCM, reuse is lower than resampling on average (0.482 versus 0.520). Under mean misspecification, resampling falls to 0.391 while factual reuse reaches 0.459, giving a reuse-minus-resampling difference of $+0.068$. So experimentally, factual reuse does reduce the degradation caused by this tested misspecification without providing a universal advantage when the SCM is accurate.

9 Discussion and Limitations

Under the tested immediate-decay schedule, ID-CAPW improves normalized AUC over factual-only PPO across all five matched seeds but not mean Final MA6 (Section 8.4), indicating a learning-trajectory rather than endpoint or computational advantage. CCW concentrates on longer observed factual correctness, whereas CAPW broadens branch support (Appendix H); factual updates may reduce sensitivity to imperfect counterfactual data, but the controlled comparison establishes only a regime interaction (Section 8.5).

Unlike model-disagreement approaches, our reliability signal is computed from factual experience before CF rollout, allowing branch selection before synthetic generation rather than filtering or truncating generated rollouts (Pan et al., 2020; Frauenknecht et al., 2024). The persistence of factual reliability rankings after counterfactual branching is supported empirically (Section 8.2) and is not theoretically guaranteed by the method; the divergence tested there does not establish robustness to large shifts. Our study is limited to one environment, a fixed heuristic defender, one learned mechanism, and three to five training seeds. Combining factual branch selection with uncertainty-aware rollout truncation is a natural direction for future work.

10 Conclusion

We studied when counterfactual experience remains useful for RL with a partially learned SCM. Factual correctness of the learned mechanism remains informative after counterfactual branching across the observed range of trajectory divergence. This makes local factual reliability actionable for counterfactual-data allocation. Against matched Uniform branch selection, CAPW improves Final MA6 in all five seeds; against factual-only PPO, it improves normalized factual-exposure AUC by 13.7% in all five seeds, although mean Final MA6 is slightly lower. More broadly, our results suggest that partially learned causal models need not be treated as uniformly trustworthy or untrustworthy: local evidence about mechanism reliability can guide where counterfactual experience is used.

AI USE STATEMENT

In this work, we used OpenAI's ChatGPT to assist with developing aspects of the conceptual framework, refining research methodology and experimental design, implementing experimental methods and interpreting experimental results. We did not use generative AI to generate synthetic datasets, formulate mathematical claims, develop or write mathematical proofs, propose or refine research hypotheses, translate material, clean or reformat datasets or support qualitative or thematic analysis. Additionally, generative AI was used to create and edit software code, prepare plotting scripts for scientific figures, summarize and review existing literature, identify relevant literature, and edit manuscript text for readability and clarity. The first author reviewed and tested all AI-assisted code against the intended experimental configurations and invariants, and executed and evaluated the experiments. AI-assisted literature and manuscript content was checked against the cited sources and underlying experimental results. We take responsibility for the final content of this work, including text, claims, and code produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The environment and implementation are described in Sections 3–7 and Appendices A–D. The appendix also gives the reliability and stochastic-treatment protocols and paired results. Training seeds are the independent replication units, and the fixed performance summaries are descriptive.

REFERENCES

Lars Buesing, Theophane Weber, Yori Zwols, Nicolas Heess, Sebastien Racaniere, Arthur Guez, and Jean-Baptiste Lespiau. Woulda, coulda, shoulda: Counterfactually-guided policy search. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJG0voC9YQ.

Timothee Cour, Ben Sapp, and Ben Taskar. Learning from partial labels. *The Journal of Machine Learning Research*, 12:1501–1536, 2011.

Zhihong Deng, Jing Jiang, Guodong Long, and Chengqi Zhang. Causal reinforcement learning: A survey. *arXiv preprint arXiv:2307.01452*, 2023.

Anca Dragan, Rachel Holladay, and Siddhartha Srinivasa. Deceptive robot motion: synthesis, analysis and experiments. *Autonomous Robots*, 39:331–345, 2015.

Bernd Frauenknecht, Artur Eisele, Devdutt Subhasish, Friedrich Solowjow, and Sebastian Trimpe. Trust the model where it trusts itself - model-based actor-critic with uncertainty-aware rollout adaption. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 13973–14005. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/frauenknecht24a.html.

Stefania Galimberti, Peter Sasieni, and Maria Grazia Valsecchi. A weighted kaplan–meier estimator for matched data with application to the comparison of chemotherapy and bone-marrow transplant in leukaemia. *Statistics in medicine*, 21(24):3847–3864, 2002.

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.

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.

Guy Lorberbom, Daniel D Johnson, Chris J Maddison, Daniel Tarlow, and Tamir Hazan. Learning generalized gumbel-max causal mechanisms. *Advances in Neural Information Processing Systems*, 34:26792–26803, 2021.

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.

Melkior Ornik and Ufuk Topcu. Deception in optimal control. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 821–828. IEEE, 2018.

Feiyang Pan, Jia He, Dandan Tu, and Qing He. Trust the model when it is confident: Masked model-based actor-critic. *Advances in neural information processing systems*, 33:10537–10546, 2020.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Roberta Raileanu, Emily Denton, Arthur Szlam, and Rob Fergus. Modeling others using oneself in multi-agent reinforcement learning. In *International conference on machine learning*, pp. 4257–4266. PMLR, 2018.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Jun Xie and Chaofeng Liu. Adjusted kaplan–meier estimator and log-rank test with inverse probability of treatment weighting for survival data. *Statistics in medicine*, 24(20):3089–3110, 2005.

Xiaopeng Yu, Jiechuan Jiang, and Zongqing Lu. Opponent modeling based on subgoal inference. *Advances in Neural Information Processing Systems*, 37:60531–60555, 2024.

Yan Zeng, Ruichu Cai, Fuchun Sun, Libo Huang, and Zhifeng Hao. A survey on causal reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(4):5942–5962, 2024.

A Environment specification

The factual objective is

$$J(\pi_\theta) = \mathbb{E}_{\tau \sim p_{\mathrm{env}}(\cdot|\pi_\theta)} \left[\sum_{t=0}^{H-1} \gamma^t r_t \right], \tag{10}$$

where π_θ is the invader policy, τ is a factual trajectory, and $\gamma = 0.99$ is the discount factor. Evaluation always executes a policy in the environment.

An episode terminates when the invader reaches its true goal, is captured by the defender, or reaches the maximum horizon $H = 200$. The transition reward is

$$r_t = r^t_{\mathrm{term}} + r^t_{\mathrm{inf}}, \qquad r^t_{\mathrm{term}} = \begin{cases} 100 + d_D^{t+1} - 50\dfrac{t+1}{H}, & \text{goal reached,} \\ -100 - d_G^{t+1} + 50\dfrac{t+1}{H}, & \text{capture or timeout,} \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where d_D^{t+1} is the post-transition invader–defender distance and d_G^{t+1} is the invader's remaining distance to the true goal. The influence term r^t_{inf} is a sparse shaping reward for sustained actions that worsen the defender's interception geometry, measured through the capture-point signal. Let p_C^t be the defender's capture point before the transition, let $d_C^t = \|p_I^t - p_C^t\|_2$ when that point exists, and define $\Delta_C^t = d_C^{t+1} - d_C^t$. Let B_t be the cumulative influence bonus already paid in the episode. A transition is eligible, $E_t = 1$, only when it is nonterminal, the invader has held the same action for at least five consecutive steps, the 15-step post-payment cooldown has expired, fewer than three payments have occurred, and $B_t < 6$. The uncapped payment is

$$b_t = \begin{cases} 1, & p_C^t \text{ exists and } p_C^{t+1} \text{ does not,} \\ 2\ \mathrm{clip}_{[0,1]}\left(\dfrac{\Delta_C^t - 10^{-3}}{50}\right), & p_C^t, p_C^{t+1} \text{ exist and } \Delta_C^t > 10^{-3}, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

The shaping term is

$$r^t_{\text{inf}} = \mathbf{1}\{E_t = 1\} \min\{b_t,\, 6 - B_t\}. \tag{13}$$

A positive payment increments the episode payment count and restarts the cooldown. Thus, the influence reward can guide deceptive behavior while remaining small relative to the terminal reward magnitude of 100. The environment and known SCM reward mechanism apply the same definition.

The policy observes $o_{I,t} = O_I(s_t)$ rather than the full simulator state s_t. Action $a_t = j \in \{0, 1, 2\}$ directs the invader towards candidate goal g_j. Each agent $A \in \{I, D\}$ follows first-order planar dynamics with position (x_A, y_A), heading β_A, speed v_A, and steering angle θ_A:

$$\dot{x}_A = v_A \cos\beta_A, \qquad \dot{y}_A = v_A \sin\beta_A, \qquad \dot{\beta}_A = \frac{v_A \theta_A}{L}, \tag{14}$$

where L is the vehicle length parameter. The invader and defender have maximum steering angles $\pi/8$ and $\pi/16$, respectively. The defender infers the invader's intended goal from its recent motion, selecting the candidate toward which the invader made the greatest distance progress since the previous step. With no prior position, the implementation uses default goal index zero. Its believed goal $g_{D,t}$ is part of the latent state but is excluded from $o_{I,t}$. Given that belief, the defender computes a capture point and steers towards it for interception, or towards the believed goal if no capture point exists. The simulator returns an 18-dimensional observation containing five invader-to-object distances and their world-frame bearings, the two agent headings, and the world coordinates of the three candidate goals. The five objects are the true goal, defender, capture point, and two false goals. If no capture point exists, its distance and bearing entries use true-goal geometry. Neither the defender's believed goal nor influence-reward bookkeeping appears in this observation. The policy transforms this observation into agent-centred features as specified in Appendix C.

The numerical dynamics use $L = 0.3$, timestep $\Delta t = 0.3$, invader speed $v_I = 1$, defender speed $v_D = 1.0001$, and capture and goal radii of one. Position is advanced by explicit Euler integration using the current heading, while the next heading follows Eq. 14 modulo 2π. Simulator coordinates are clipped to $[-50, 50]$ on each axis. Initial cases are generated in $[0, 50]^2$ with margin five. The defender starts at $(25, 25)$, the invader is sampled uniformly along the perimeter of $[5, 45]^2$, and both initial headings are uniform on $[0, 2\pi)$. Each goal is sampled uniformly in the inner square subject to being closer to the defender than the invader. Goal triples with triangle area below 100 are rejected. Goal index zero is the true goal. The NumPy case-generator seeds are 100 for training and 1729 for evaluation, with cases stored in groups of 1000.

For a believed goal, the defender predicts an invader ray $p^t_I + \lambda u^t_I$ toward that goal and chooses the smallest nonnegative λ satisfying $\|p^t_I + \lambda u^t_I - p^t_D\|_2 / v_D = \lambda / v_I$. The implementation solves the resulting quadratic. If the valid point lies beyond the believed goal or no nonnegative solution exists, the defender steers toward the believed goal instead.

Table 1: Environment and evaluation settings used in the reported experiments.

Group	Parameter	Value
Dynamics	Horizon / timestep	200 / 0.3
Dynamics	Invader / defender speed	1 / 1.0001
Control	Invader / defender steering limit	$\pi/8$ / $\pi/16$
Termination	Goal / capture radius	1 / 1
Evaluation	Held-out cases	evaluation indices 1000–1499
Evaluation	Cases / policy action	500 / deterministic
Metrics	Endpoint statistic	chronological Final MA6

B MOA REPRESENTATION AND TRAINING

The MOA uses an 18-dimensional input. It concatenates defender position rotated into the invader's local frame (two coordinates), relative defender heading (sine and cosine), the invader's previous-step displacement in its local frame (two coordinates), and four features for each of three candidate goals. Each goal block contains a rotated goal displacement, its Euclidean distance, and the one-step change in that distance. The previous-step displacement is zero at the start of an episode. The deployed invader policy does not receive the MOA prediction.

For each candidate believed goal, the specified defender rule generates a response representation containing next defender position and heading, capture-point validity and location, and the choice

to move toward the believed goal or capture point. Candidate goals are response-equivalent when the maximum absolute difference between their encoded responses is at most 10^{-4}. The first term of Eq. 5 places probability on the entire response-consistent set. The second is the probability-weighted response discrepancy. The implementation clamps response-consistent probability mass below at 10^{-8} before taking its logarithm. The MOA architecture and optimizer are summarized with the policy-update profiles in Table 2. It updates after each source unit.

In Eq. 5, $\Delta(s, g)$ is the mean squared difference across the encoded candidate and factual response components, with no additional discrepancy cap. Both loss terms have unit coefficients. Rollouts use the highest-probability believed goal rather than posterior sampling. The same MOA training profile is used in the learned-MOA studies reported here.

C CG-PPO IMPLEMENTATION

Policy representation. All compared policies use the deterministic agent-centred flat extractor (`agent_flat`), with additional defender-to-object features disabled. It rotates object displacements into the invader's frame and flattens five six-component tokens, ordered as defender, true goal, two false goals, and capture point, into 30 features. Every token contains local x, y displacement. The defender token also contains cosine and sine of relative heading. Remaining token entries are zero, and absolute goal coordinates are not forwarded to the network. There is no attention layer or learned feature projection. Separate actor and critic networks each have two 64-unit tanh layers, followed by a three-action categorical head and a scalar value head, respectively. Observations and rewards are not adaptively normalized.

Policy updates. The target, reference, and planner policies have distinct roles. Factual actions are generated by the reference policy before optimization and receive importance weight one. Counterfactual actions are generated by π_λ and corrected toward π_{ref} with

$$w_t = \exp[\mathrm{clip}(\log \pi_{\mathrm{ref}}(\hat{a}_t \mid \hat{o}_t) - \log \pi_\lambda(\hat{a}_t \mid \hat{o}_t), -\kappa, \kappa)]\,, \tag{15}$$

where $\kappa = 2.5$. These are per-step weights, without trajectory-level averaging of log ratios or additional MOA-confidence loss weights.

For minibatch $\mathcal{B}$, define $\mathbb{E}_w[u_t] = \sum_{t\in\mathcal{B}} w_t u_t / \max(\sum_{t\in\mathcal{B}} w_t, 10^{-8})$ and $\rho_t(\theta) = \pi_\theta(a_t \mid o_t)/\pi_{\mathrm{ref}}(a_t \mid o_t)$. The actor maximizes $\mathbb{E}_w[\min(\rho_t \hat{A}_t, \mathrm{clip}(\rho_t, 1-\epsilon, 1+\epsilon)\hat{A}_t)]$. The value loss is one half of the weighted Smooth-L1 error, with transition point one, between $V(o_t)$ and the generalized-advantage-estimation (GAE) return target. The total minimized loss adds 0.5 times this value loss to the negative actor objective. Value clipping is disabled. GAE uses $\gamma = 0.99$ and $\lambda = 0.95$. Terminal transitions have zero bootstrap value, while nonterminal retained boundaries bootstrap from the reference value function. Advantages are standardized over valid transitions in the complete update unit, with denominator epsilon 10^{-8}, before minibatch shuffling.

The numerical model and optimizer settings are collected in Table 2. Actor and critic use the same learning rate within each PPO profile. An update stops before processing a minibatch whose weighted approximate KL, $\mathbb{E}_w[\exp(\log \rho_t) - 1 - \log \rho_t]$, exceeds 1.5 times the target. These profiles apply to the main ID-CFF experiment, CF-only selector comparison, controlled factual-PPO comparison, R4 schedule comparison, reliability validation, and stochastic study. Factual-only PPO uses only the factual profile.

Sources and rollout batches. Each source contains 50K retained factual transitions. It supplies MOA supervision and branch candidates, and supplies factual PPO data when that update is enabled. The MOA is trained before correctness annotation, and annotations remain fixed during the source's CF updates. Trajectories are visited in shuffled round-robin cycles, giving equal trajectory allocation, with the selector determining the branch index within each trajectory. Only the current source is used. The CF-only selector and stochastic studies execute four 50K CF units per source without factual PPO. The controlled no-factual-PPO condition retains the ID source and MOA lifecycle but omits the factual policy update.

A CF update unit contains 50K valid transitions, not a fixed number of rollouts or collector requests. Adaptive collection chunking changes throughput only; it does not change the retained update-unit

Table 2: Learned-model and policy-optimization settings. A dash denotes a non-applicable quantity.

Parameter	MOA	Factual PPO	CF PPO
Hidden layers	$128, 128$	$64, 64$ actor/critic	same policy
Activation	ReLU	tanh	tanh
Learning rate	10^{-4}	5×10^{-4}	3×10^{-4}
Epochs	3	up to 6	up to 6
Minibatch	96	64	64
Optimizer	Adam	Adam	Adam
Target KL / PPO clip	–	0.03 / 0.15	0.03 / 0.15
Discount / GAE λ	–	0.99 / 0.95	0.99 / 0.95
Entropy / value coefficient	–	0 / 0.5	0 / 0.5
Gradient norm / Adam ϵ	–	$0.5 / 10^{-5}$	$0.5 / 10^{-5}$
Buffer capacity	8192	–	–
Response tolerance	10^{-4}	–	–

budget or PPO minibatch size. Chunks accumulate until the unit budget is met, excess transitions are discarded, and rollout tails are not reused across units. Individual CF rollouts stop at simulated termination or the remaining episode horizon of 200, with no additional depth cap. The planner is fixed within a CF unit.

Planner bank. The bank starts with the initial policy and adds a target-policy snapshot after each policy-update unit. It retains the 20 newest snapshots and up to 16 older snapshots ranked by score. Planner selection is restricted to the recent 20. When scored candidates exist, a probability-0.30 branch samples them using weights proportional to $(s_i - \min_j s_j + 10^{-6})^2$, mixed with 0.20 uniform probability. Otherwise, selection uses recency weights $\exp(-0.25 a_i)$, where a_i is age in snapshots. Within this recency branch, probability 0.25 restricts selection to recent unscored snapshots when available. A snapshot's score is the running mean of its CF-unit mean undiscounted retained-rollout return. Held-out evaluation scores are not used for planner selection. There is no additional holding period or anti-crowding penalty.

Selector implementation. CCW uses the linear weight in Eq. 6, equivalent to exponent one. CAPW fits its unsmoothed product-limit curve once per annotated source, with zero risk mass contributing zero hazard. No CCW epsilon offset is added to CAPW weights. Recorded diagnostics show no CAPW uniform-fallback events across the five main ID-CAPW runs, so unused edge-case fallback settings are omitted here.

D Evaluation and experiment inventory

The main ID-CFF experiment, CF-only selector comparison, and controlled factual-PPO comparison use seeds 0, 50K, 100K, 150K, and 200K, where K denotes 1000. The reliable-SCM motivation, reliability validation, and R4-CAPW schedule comparison use seeds 0, 50K, and 100K. The stochastic study uses seeds 300K, 350K, 400K, 450K, 500K, and 550K. Training seeds initialize the learning random-number generators and select the starting index in the training-case sequence. The reliability-selector generator uses the training seed XOR `0x6D2B79F5`.

Table 3: Training cohorts and experimental budgets. Factual exposure denotes retained source transitions; CF exposure denotes optimized counterfactual transitions.

Cohort	Seeds	Factual	CF	Factual PPO
CF-only selector	0, 50K, 100K, 150K, 200K	2M	8M	No
Reliability validation	0, 50K, 100K	3M	12M	No
Main ID CFF	0, 50K, 100K, 150K, 200K	3M	4.75M	Yes
Controlled ID comparison	0, 50K, 100K, 150K, 200K	3M	4.75M	Yes/No
R4 schedule comparison	0, 50K, 100K	3M	12M	Yes
Stochastic study	300K–550K, six seeds	3M	12M	No

Policies are evaluated with deterministic action selection on 500 cases, indices 1000–1499 of the evaluation corpus. Case identity is verified in the recorded endpoint artifacts for the main ID-CFF and stochastic experiments. The CF-only selector runs retained the panel configuration but not explicit case IDs, so retrospective identity is not independently verified for that cohort. Evaluation

does not train the policy or MOA. The CF-only selector comparison uses 40 factual sources (2M retained factual transitions) and 8M optimized CF transitions. The reliability validation uses 60 sources and 12M CF transitions. The main ID-CFF and R4 studies use 60 sources. Removing factual PPO in the controlled comparison preserves the source-exposure axis.

For the immediate-decay (ID) schedule, each factual source $j \in \{1, \ldots, 60\}$ contains 50K retained factual steps and contributes $50{,}000\, r_j$ CF-step credit, with $r_j = 4 \times 2^{-50{,}000 j / 950{,}000} = 4 \times 2^{-j/19}$. Thus, decay starts at the first source and the credit ratio halves every 950K factual steps. The half-life approximately matches the cumulative exposure of the earlier delayed schedule over 3M factual steps while changing the timing. At each source, inherited residual credit is added to new credit, complete 50K CF units are executed from that source, and the remainder carries forward. ID yields 95 CF units (4.75M transitions), whereas R4 executes four per source, giving 240 units (12M transitions).

Table 4: Active source, selector, and scheduler settings.

Quantity	Setting
Retained factual source / CF update unit	50K / 50K transitions
Fixed-source reuse	four CF units per source where prescribed
CCW	$L_{\max} = 40$, $\epsilon = 0.05$, $\alpha = 1$
CAPW	unsmoothed product-limit midpoint percentile
ID credit ratio at source j	$4\,2^{-j/19}$, with global residual carry
R4 ratio	four complete CF units per factual source
CF horizon	termination or remaining episode horizon
Planner/reference log-ratio clamp	2.5

CG-PPO is evaluated after each 50K policy-update unit. For the main factual-exposure comparison, select the latest chronological evaluation at each repeated 50K source boundary from 50K through 3M, then use trapezoidal integration divided by 2.95M. Chronological Final MA6 is computed over the last six raw evaluations before this deduplication. The CF-only plotted grid is 0.2M–8M optimized CF steps in 0.2M increments, with AUC normalized by its 7.8M-step width. AUC values are compared only within matched exposure domains. Training seed, rather than evaluation case, factual source, or trace, is the unit of learning replication. The maintained protocol was fixed for post-hoc descriptive analysis, not prespecified confirmatory inference.

E STOCHASTIC TREATMENT

The separate six-seed stochastic comparison fixes the CCW rule and uses CF-only policy updates, with 3M retained factual-source steps and 12M optimized CF steps per run. The accurate-model reuse-minus-resampling mean differences are -0.038 in source-exposure AUC and -0.054 in Final MA6. Under the tested mean misspecification they are $+0.068$ and $+0.172$. These are paired descriptive results, not effects of the final ID-CAPW pipeline.

The four treatment summaries plotted in Figure 5(f) are given in Table 5. Each cell uses the same six matched training seeds: 300K, 350K, 400K, 450K, 500K, and 550K. Error bars in the figure are one sample standard deviation across these seeds and are descriptive rather than confidence intervals.

Table 5: Normalized factual-exposure AUC in the stochastic experiment. Values are mean $\pm$ one sample SD across six matched seeds.

Rollout handling	Correct SCM	Mean-misspecified SCM
Resampling	0.520 ± 0.066	0.391 ± 0.022
Factual reuse	0.482 ± 0.075	0.459 ± 0.043

Independent invader and defender heading disturbances follow $\mathcal{N}(0.030, 0.100^2)$ in the environment. The accurate SCM uses the same law, while the misspecified SCM uses $\mathcal{N}(-0.030, 0.100^2)$. Each disturbance is added to the computed next heading modulo 2π, after deterministic motion. Factual reuse takes retained disturbances where the matched suffix supplies them and uses fresh SCM draws beyond that support. Resampling uses fresh SCM draws throughout. The evaluation disturbance master seed is 4040001, with case-matched disturbance schedules across treatments. Reuse therefore reduces but does not eliminate dependence on the SCM exogenous law.

Table 6: Per-agent heading-disturbance laws in the stochastic experiment.

Quantity	Environment	Correct SCM	Mean-misspecified SCM
Mean	0.030	0.030	−0.030
Standard deviation	0.100	0.100	0.100

F RELIABLE-SCM MOTIVATION

The three-seed fixed-ratio reference comparison has mean normalized AUC and Final MA6 of 0.413 and 0.520 for SB3 PPO, 0.449 and 0.655 for factual PPO, 0.494 and 0.622 for learned-SCM CG-PPO, and 0.584 and 0.671 for the reliable-SCM reference. These are a separate cohort from the main five-seed mixed-learning experiment. The CG-PPO arms use the factual and CF optimizer profiles in Appendix C, with the CF arms using R4 and uniform branch selection. The reliable reference supplies the true defender-belief mechanism.

The calibrated baseline uses Stable-Baselines3 2.7.1 PPO with the profile in Table 7. Its value objective is the SB3 squared-error loss, rather than CG-PPO's Smooth-L1 loss, and SB3 normalizes advantages within minibatches.

Table 7: Calibrated SB3 PPO profile used in the reliable-SCM motivation comparison.

Parameter	Value	Parameter	Value
Environments / rollout	1 / 2048	Learning rate	3×10^{-4}
Epochs / minibatch	6 / 64	PPO clip / target KL	0.10 / 0.03
Discount / GAE λ	0.99 / 0.95	Entropy / value coefficient	0 / 0.5
Gradient norm / Adam ϵ	0.5 / 10^{-5}	Value clipping	disabled
Architecture	agent-flat, 64–64	Evaluation interval	50K factual steps

G R4 SCHEDULE COMPARISON

The matched three-seed sustained-R4 schedule comparison completed 3M factual and 12M optimized counterfactual steps per run. Figure 5(e) compares ID-Uniform, ID-CCW, ID-CAPW, R4-CCW, and R4-CAPW on the common seeds 0, 50K, and 100K. Its x-axis is factual environment exposure from 50K through 3M steps and its y-axis is chronological Win-rate MA6. For each seed, MA6 is calculated over chronological evaluations before retaining the latest evaluation at each factual-step boundary. Curves show the three-seed mean; the shaded region is ± 0.25 sample SD for visual legibility and is not a confidence interval.

Table 8: Matched three-seed selector-by-schedule summary supporting Figure 5(e). Values are mean $\pm$ one sample SD. AUC uses the maintained normalized factual-exposure convention and is not obtained by integrating the displayed MA6 curves.

Treatment	Normalized factual-exposure AUC	Final MA6
ID-Uniform	0.479 ± 0.036	0.569 ± 0.010
ID-CCW	0.373 ± 0.182	0.450 ± 0.212
ID-CAPW	0.489 ± 0.018	0.615 ± 0.015
R4-CCW	0.394 ± 0.174	0.426 ± 0.217
R4-CAPW	0.518 ± 0.035	0.541 ± 0.126

Table 9 gives the paired CAPW-minus-CCW differences under R4, with chronological Final MA6 calculated from the last six evaluations.

The metric-specific negative seeds differ, and seed 50K strongly influences both means. The matched selector-by-schedule mean AUC interaction is only $+0.0074$ with mixed signs, so this comparison does not establish a material schedule change in the selector effect.

H SELECTOR DIAGNOSTICS

For a saved factual candidate pool with sampling probabilities p_i, define relative effective support as $q_{\text{eff}} = (1/\sum_i p_i^2)/N$, where N is the number of candidate states. The same-pool reconstruction applies both selectors to each saved pool, then averages source values within runs and the five run values within treatment. On the ID-CAPW pools, CAPW has $q_{\text{eff}} = 0.6466$ and selected observed correctness length 16.35, versus 0.5669 and 20.16 for CCW. On the ID-CCW pools, the

Table 9: Paired R4 selector effects. Positive values favour CAPW.

Training seed	Normalized AUC difference	Final MA6 difference
0	$+0.0307$	-0.1470
50K	$+0.3598$	$+0.4643$
100K	-0.0184	$+0.0263$
Mean	$+0.1241$	$+0.1146$

corresponding values are 0.6423 and 16.16 for CAPW versus 0.5640 and 19.86 for CCW. These comparisons describe sampling laws on fixed pools. Pools encountered by separately trained policies are treatment dependent, so the diagnostics do not identify a learning mediator or establish greater exploration.

I COUNTERFACTUAL RELIABILITY VALIDATION

Uniformly sampled branches are grouped into within-seed factual-R_{40} quintile bins, dropping duplicate quantile boundaries. The highest and lowest bins are compared within each factual source, and informative sources receive equal weight. The landmark analysis retains branches with a correct shared first response and at least two retained CF transitions. Subsequent mismatches are events and retained-rollout ends are right censored. Post-branch horizons $1, 5, 10, 20, 40$ correspond to original response steps $2, 6, 11, 21, 41$.

Of 516,822 eligible traces, 443,498 pass the measurable-divergence check used in Section 8.2, which requires mean first-step position or heading difference above 10^{-8}. Restricting to these traces gives high-minus-low survival gaps of 0.546 at horizon 1 and 0.226 at horizon 40, with all 15 seed-by-horizon contrasts positive.

For Figure 4(b), factual–counterfactual trajectory divergence is measured over the aligned retained post-branch prefix. At each timestep, normalized invader–defender positional divergence and normalized heading divergence are averaged, then averaged over the shared prefix. The common branch state is excluded. The resulting quantity is dimensionless and should not be interpreted as raw positional distance. The five frozen divergence quintiles span $[0, 0.4305]$, with median divergence increasing from 0.0099 in the lowest quintile to 0.1157 in the highest. These values characterize the relatively local counterfactual interventions generated under the CG-PPO policy-mismatch controls rather than arbitrary large distribution shifts.

The high-minus-low survival contrast remains positive in every divergence quintile at CF timesteps 5, 10, 20, and 40. All 15 seed-by-divergence cells contain at least 56 informative sources. Excluding branches with fewer than five aligned post-branch states produces nearly unchanged results and preserves positive contrasts in the highest-divergence quintile. Because divergence is realized after branching, these comparisons establish predictive reliability transfer across the observed divergence range rather than a causal effect of divergence or robustness to arbitrarily distant interventions.