CCPL: STRESS-TESTING DELAYED-COST ATTRIBUTION UNDER SAFETY CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Constrained reinforcement learning often pairs an observed safety cost with the most recent action. When consequences arrive after stochastic delays, this temporal pairing can send the learning signal to the wrong decision and leave the responsible decision uncorrected. We introduce Causal Consequence-Penalized Learning (CCPL), an auditable delayed-cost attribution layer for constrained reinforcement learning. Its central mechanism is an explicit posterior-to-penalty interface: a label-free operational posterior over source lags combines a learned delay prior with a consequence-model likelihood, assigns the identifiable share of a cost to earlier actions, and retains unresolved responsibility as a state-level risk signal. For jointly caused events, a Shapley split distributes the explainable share across candidate actions. Thus CCPL separates uncertainty about which action caused a cost from uncertainty about its magnitude. Across five seeds in a bimodal delayed-cost benchmark, the hindsight assignment reduced mean consequence by 0.278 with a paired 95% interval of $[-0.474, -0.018]$ and increased constraint satisfaction from 0.84 to 0.98, while the reward difference was not statistically distinguishable from zero. With source-lag labels withheld from the operational assignment path, attribution overlap was 0.17–0.20 across three delay profiles; a uniform split reached higher raw overlap because responsibility was diffuse, but did not translate into better policy performance. A simulator intervention diagnostic with $n = 1200$ found that responsibility shares tracked measured intervention effects with Pearson $r = 0.99$, compared with $r = 0.43$ for path-Shapley and $r = 0.00$ for visible-step attribution. The learned delay distribution recovered a known constant delay only after extended training, reaching a predicted mean of 9.9992 for a true delay of 10. The gains are conditional: the mechanism improves safety in the binding delayed regime studied here, while other held-out conditions do not separate the methods. The action-level consequence contrast uses simulator structural-equation supervision for training and intervention validation; label-free refers specifically to the operational assignment path, which receives no source-lag or SCM label at assignment time. We therefore present CCPL as a stress-tested attribution mechanism and evaluation framework for delayed safety costs, not as a general causal-identification procedure or a guarantee for full neural training.

1 INTRODUCTION

Constrained reinforcement learning (RL) seeks a policy that maximizes utility while limiting an expected safety consequence. Its training signal is frequently treated as if the observed cost and the most recent action were causally paired – reasonable for an immediate cost, not when a consequence passes through a delayed pipeline.

Suppose an action at time t initiates a process whose consequence is observed at $t + \tau_t$, after actions $a_{t+1}, \ldots, a_{t+\tau_t-1}$ have already occurred. Assigning the observation to $a_{t+\tau_t}$ is a temporal attribution error: the policy may avoid actions correlated with a dangerous state while never correcting the action that changes the consequence.

This creates a practical need beyond a better scalar penalty. A practitioner must be able to answer three questions after a safety event: which earlier decision was plausibly responsible, how uncertain is that attribution, and when should the learner avoid assigning blame to any single action? We design

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.

CCPL for this audit loop. We record a source posterior and its entropy, turn concentrated evidence into an action-level correction, and retain unresolved risk as an explicitly state-level signal. Thus CCPL is useful when safety feedback is delayed and accountability matters: it makes the credit decision inspectable, exposes when the decision is unreliable, and can be used as an attribution layer around an existing constrained-RL controller.

CCPL makes this failure mode explicit with five connected pieces: a delay model and effective discount, replay-time alignment of an emitted consequence to an earlier labeled action, an ICN estimating an action-centered contrast, a state-conditioned bounded safety weight, and separate reward and consequence critics. The key new mechanism operates without a source label: a hindsight posterior combines the learned delay prior with the ICN's consequence likelihood, and its entropy decides whether unresolved cost goes to a state rather than a falsely accused action. Jointly caused events are split by a Shapley share (Shapley, 1953); an uncertainty/OOD gate adds bounded conservative margins.

We distinguish model supervision from operational assignment. The operational posterior assigns observed costs without source-lag or SCM labels, using the learned delay prior and consequence model. In controlled experiments, simulator SCM labels train and calibrate the ICN's action-level contrast and provide reference targets for intervention validation. Thus, "label-free" describes the assignment path, not all model training. The state-conditioned multiplier is a training heuristic, and the formal delay analysis concerns a fixed-kernel tabular update.

Our novelty is the closed-loop coupling: one label-free posterior records source attribution, trains delay inference, and routes unresolved mass to a state residual, making CCPL an auditable attribution layer that can wrap constrained controllers subject to environment-specific calibration, rather than another standalone penalty or hindsight classifier.

We make three contributions:

1. a label-free-at-assignment hindsight attribution layer: a posterior over source lags from delay prior and consequence likelihood, with an entropy-gated action/state split that keeps unresolved risk at the state rather than inventing a precise action label;
2. conditional safety evidence: paired five-seed evidence for where attribution converts to safety (bimodal delay under binding constraints) versus where it does not, with cost-weighted provenance, label-noise and residual controls reported as-is; and
3. intervention and cross-environment validation: a `do()`-style check against measured simulator intervention effects, plus the same posterior as a drop-in patch for a third-party Lagrangian baseline and a four-condition GUARD-style protocol with an oracle-anchored timing reference.

We evaluate provenance overlap separately from usefulness. In our benchmark, uniform allocation can achieve higher overlap because responsibility is diffuse across many candidate actions. The label-free posterior instead provides a source-sensitive safety signal whose usefulness is evaluated through constraint satisfaction and intervention measurements.

Together, these contributions target an attribution bottleneck rather than a universal controller replacement. Section 7 evaluates both the attribution signal itself and the narrower question of when it improves constraint satisfaction.

2 Background and Related Work

2.1 Constrained RL

CMDPs pair a reward objective with a constraint budget (Altman, 1999). The baseline family spans trust-region updates (Achiam et al., 2017), Lyapunov reformulations (Chow et al., 2018), Lagrangian/primal–dual methods (Tessler et al., 2019; Chow et al., 2017), first-order and penalty approximations including our CPO-FO baseline (Zhang et al., 2020; Yang et al., 2022; Zhang et al., 2022; Liu et al., 2020; Milošević et al., 2025), budget state augmentation (Sootla et al., 2022), and PPO as the unconstrained reference (Schulman et al., 2017). None address *when* a cost should be attributed to a past action; CCPL modifies *which* action receives the penalty, making it compatible with most of this family. The Omnisafe library standardizes many of these baselines (Ji et al., 2024);

Safety Gym and Safety-Gymnasium provide the corresponding environments (Ray et al., 2019; Ji et al., 2023).

2.2 Delayed and anonymous reward

Delayed observation relates to partial observability, where histories summarize information absent from the current observation; a well-tuned recurrent policy is a strong general POMDP baseline (Hausknecht & Stone, 2015; Ni et al., 2022), motivating our finite history summary h_t. Classical delay-aware RL assumes a known fixed lag with exponential state augmentation (Katsikopoulos & Engelbrecht, 2003); CCPL instead infers an unknown stochastic lag per cost without augmentation. Regret analysis for delayed, composite, and partially anonymous reward (Mondal & Aggarwal, 2023) is the closest prior formal treatment, though without causal attribution. RUDDER decomposes trajectory return into per-step proxy rewards (Arjona-Medina et al., 2019), addressing delayed *reward* without an explicit lag posterior.

2.3 Causal credit assignment

The causal notation follows structural causal models (Peters et al., 2017; Schölkopf et al., 2021). CCPL uses the interventional estimand only in the controlled simulator, where counterfactual labels come from known structural equations – useful for validating agreement with the simulator's known interventional contrast. The operational hindsight path does not require those labels; it is evaluated separately from the privileged calibration path. Hindsight Credit Assignment reweights past decisions by a learned hindsight classifier (Harutyunyan et al., 2020; Zhang et al., 2023); counterfactual baselines solve *who* across agents rather than *when* across time. CCPL targets delayed *constraint cost* with a co-trained generative delay process, a conserved cost split, and separated source/magnitude uncertainty: an integrated, auditable mechanism for delayed constraint cost.

Unlike Shapley, HCA, or RUDDER-style redistribution, CCPL jointly infers the hidden stochastic delay and allocates delayed constraint cost while preserving an uncertainty residual for unresolved responsibility.

2.4 Uncertainty estimation

The uncertainty head's Laplace-style objective (Section E.5) follows the general approach of learning a predictive variance jointly with a point estimate (Kendall & Gal, 2017). We use the uncertainty estimate only as a bounded conservative margin at the policy's action-selection gate, not as a calibrated confidence interval (Section E.5).

3 Problem Formulation

Let a finite-horizon CMDP have state $s_t \in \mathcal{S}$, discrete action $a_t \in \mathcal{A}$, transition kernel P, reward r_t, non-negative consequence c_t, discount $\gamma \in (0, 1)$, horizon T, and budget d. For policy π,

$$J_r(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} \gamma^t r_t \right], \qquad J_c(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} \gamma^t c_t \right]. \tag{1}$$

The target problem is $\max_\pi J_r(\pi)$ subject to $J_c(\pi) \leq d$.

3.1 Delayed consequence observation

Let x_t denote physical state and z_t pending consequence state. A useful abstraction is

$$x_{t+1} \sim P(\cdot \mid x_t, a_t), \tag{2}$$
$$z_{t+1} = F(z_t, x_t, a_t, \xi_t), \tag{3}$$
$$c_t = G(z_t, \eta_t), \tag{4}$$

where ξ_t, η_t are exogenous disturbances. The augmented state (x_t, z_t) is Markov when z_t is observed. When it is hidden, the learner uses a history h_t as a finite summary. The issue is not that physical

dynamics violate the Markov property; it is that the emitted consequence is not necessarily caused by the action active when it is observed.

Let $\tau_t \in \{0, \dots, K\}$ denote the delay associated with a consequence and $p_\varphi(k \mid h_t)$ its learned conditional distribution, with delay weight

$$\bar{\gamma}(h_t) = \sum_{k=0}^{K} p_\varphi(k \mid h_t)\gamma^k. \tag{5}$$

This is a delay weight, not a responsibility estimate, which comes from the aligned replay label or SCM contrast. Figure 3 (Appendix B) illustrates the alignment this corrects for.

4 CCPL METHOD

The implementation data flow, the ICN's two-head factorization, and the alignment mechanism are diagrammed in Appendix B (Figures 2, 4, 3).

4.1 REPLAY ALIGNMENT AND DELAY LEARNING

The agent stores transitions, hidden states, and the finite causal history. When the environment exposes `actual_tau`, CCPL writes the emitted consequence into the earlier transition (privileged oracle/diagnostic condition only). In the label-free hindsight condition this path is disabled and the posterior in Eq. (12) supplies soft alignment. The delay network is trained with weighted categorical cross-entropy:

$$\mathcal{L}_{\text{delay}}(\phi) = -\frac{1}{\sum_i w_i} \sum_i w_i \log p_\phi(\tau_i \mid h_i), \qquad w_i \geq 0. \tag{6}$$

If too few observed delay labels exist, a bounded online prior warm-starts updates; this fallback claims no correctness for the prior.

4.2 IMPLEMENTATION-ALIGNED CRITIC TARGETS

The reward critic keeps the ordinary one-step target:

$$y_t^r = r_t + \gamma(1 - d_t)Q_r^-(s_{t+1}, a_{t+1}). \tag{7}$$

The consequence critic uses the positive clipped attributed consequence:

$$y_t^c = \bar{\gamma}(h_t)[\Delta C_t]_+ + \gamma(1 - d_t)Q_c^-(s_{t+1}, a_{t+1}). \tag{8}$$

The multiplier does not appear in either critic target; it enters when actions are scored, keeping reward and consequence estimates independently inspectable as the safety weight changes.

4.3 ACTION-CENTRED CONSEQUENCE ATTRIBUTION

For a reference action a_0, the simulator supplies the interventional estimand

$$\Delta C(s, a, h) = \mathbb{E}\left[C \mid \text{do}(A_t = a), s, h\right] - \mathbb{E}\left[C \mid \text{do}(A_t = a_0), s, h\right]. \tag{9}$$

Figure 5 (Appendix B) illustrates why this is an interventional rather than observational contrast.

The ICN predicts a total consequence and a same-state action baseline,

$$\widehat{\Delta C}_\psi(s, a, h) = f_{\text{total},\psi}(s, a, h) - f_{\text{base},\psi}(s, h). \tag{10}$$

The state is retained in both heads; the baseline omits the current action so that it represents action-marginal consequence at the same state and history. For valid labels, the empirical weighted squared error for the contrast is

$$\widehat{\mathcal{R}}_{\text{contrast}}(\psi) = \frac{1}{\sum_i w_i} \sum_{i=1}^{B} w_i \left(\widehat{\Delta C}_\psi(s_i, a_i, h_i) - \Delta C_i\right)^2. \tag{11}$$

The implementation fits the total and baseline heads separately rather than minimizing this contrast error directly; Appendix E.2 specifies those losses. It also trains a positive uncertainty scale σ_ψ with a Laplace-style error objective, used only to add a bounded margin at the penalty gate – never to shrink a penalty under uncertainty.

4.4 HINDSIGHT POSTERIOR AND ENTROPY-GATED RESIDUAL

No SCM-supervised ICN is needed at assignment time. For an observed cost without source label, CCPL forms a posterior over candidate source lags from the learned delay prior and ICN likelihood:

$$w_{t,\tau} = \frac{p_\varphi(\tau \mid h_t)\, \ell(c_t \mid \widehat{\Delta C}_{t,\tau}, \sigma_{t,\tau})}{\sum_{u \in \mathcal{T}_t} p_\varphi(u \mid h_t)\, \ell(c_t \mid \widehat{\Delta C}_{t,u}, \sigma_{t,u})}, \qquad \tau \in \mathcal{T}_t. \tag{12}$$

We use the positive-scale Laplace likelihood

$$\ell(c \mid \mu, \sigma) = \frac{1}{2\sigma} \exp\left(-\frac{|c-\mu|}{\sigma}\right), \qquad \sigma > 0. \tag{13}$$

with μ set to the positive-clipped contrast $[\widehat{\Delta C}_{t,\tau}]_+$, not the total predicted consequence. This is a ranking model, not a generative one (c_t is total cost, μ a marginal contribution); only the *relative* ordering across lags matters for the weights, and the overlap/TAE and intervention measurements test whether that ordering is useful. For a jointly caused event, define the non-negative candidate contributions and the saturating coalition game

$$x_{t,\tau} = p_\varphi(\tau \mid h_t) \left[\widehat{\Delta C}_{t,\tau}\right]_+, \qquad v_t(S) = \min\left\{c_t, \sum_{\tau \in S} x_{t,\tau}\right\}. \tag{14}$$

With $m = |\mathcal{T}_t|$, the exact Shapley value is

$$\phi_{t,\tau} = \sum_{S \subseteq \mathcal{T}_t \setminus \{\tau\}} \frac{|S|!(m-|S|-1)!}{m!} \left[v_t(S \cup \{\tau\}) - v_t(S)\right]. \tag{15}$$

The implementation uses exact subset evaluation for small candidate sets and permutation sampling for larger ones, then normalizes to assignment weights, so the credited costs $c_t w_{t,\tau}$ sum to the observed undiscounted cost (a property of the attribution record, not of the separate penalties in Eq. (17)). Redistributing cost across source times can therefore change a discounted learning target; conservation does not imply discounted-target invariance. The posterior also supplies soft targets for delay learning in an EM-style update: a posterior E-step followed by a weighted delay-model update. The the hindsight condition reads only the observed scalar cost, never the source-lag label. Algorithm 2 lists every step.

The ICN scale σ describes uncertainty in consequence magnitude; it is not temporal-source confidence. We therefore compute posterior entropy $H_t = -\sum_{\tau \in \mathcal{T}_t} w_{t,\tau} \log w_{t,\tau}$ and define the normalized entropy by

$$\widetilde{H}_t = \begin{cases} H_t / \log|\mathcal{T}_t|, & |\mathcal{T}_t| > 1, \\ 0, & |\mathcal{T}_t| = 1. \end{cases} \tag{16}$$

Let $w_{\max} = \max_\tau w_{t,\tau}$ and $\tau^\star$ be the corresponding lag. The action-level penalty and unresolved state residual are

$$\Pi_{\text{action},t} = w_{\max}\, [\widehat{\Delta C}_{t,\tau^\star}]_+, \qquad \Pi_{\text{state},t} = \mathbf{1}\{\widetilde{H}_t \geq \eta\}(1 - w_{\max}) c_t. \tag{17}$$

The residual is fitted from the posterior-selected risky state as an action-independent penalty, not action-source attribution, so the two terms are not asserted to sum to the observed cost. The default gate is $\eta = 0.25$; `hindsight-no-residual` removes Π_{state} and `hindsight-no-split` replaces the final allocation by a one-hot posterior argmax.

4.5 ADAPTIVE SAFETY SELECTION

The multiplier is bounded by

$$\lambda_\phi(s) = \lambda_{\max}\, \text{sigmoid}(f_\phi(s)), \qquad 0 \leq \lambda_\phi(s) \leq \lambda_{\max}. \tag{18}$$

The implementation trains its target from global episode violation, local attributed consequence, and recent hit frequency – an adaptive heuristic for the single aggregate constraint, not a closed-form statewise dual.

At action selection, the policy combines reward value, the consequence critic, and the ICN estimate:

$$a_t = \arg\max_a \{Q_r(s_t, a) - \Pi_{\mathrm{ICN}}(s_t, a, h_t) - \lambda_\phi(s_t)\Pi_{Q_c}(s_t, a)\}. \tag{19}$$

Here Π_{ICN} contains the candidate-specific action penalty and the entropy-gated state residual from Eq. (17), broadcast across actions because it represents unresolved state risk rather than a source-action claim. Both penalty terms are clipped, and the policy shifts from the direct ICN penalty toward the learned consequence critic as training proceeds. The HallucinationGate adds bounded uncertainty and OOD margins for later audit.

5 Formal Analysis

The result below is about the fixed-kernel tabular map of Equation (8) only – not the nonlinear optimizer, the learned history encoder, or online SCM calibration.

Theorem 1 (Fixed-kernel contraction of the consequence critic)**.** *Let $x = (s, h)$ be the augmented state containing the physical state and the finite history used by the implementation. Let the augmented transition kernel $P_{\mathrm{aug}}(\cdot \mid x, a)$ and $p_\varphi(\tau \mid h)$ be fixed. Assume bounded ΔC and $0 < \gamma < 1$. Let $a^*(x')$ be any fixed deterministic action-selection rule; a reward-greedy rule is one example. Define*

$$(\mathcal{T}_c Q)(x, a) = \bar{\gamma}(h)[\Delta C(x, a)]_+ + \gamma \mathbb{E}_{x' \sim P_{\mathrm{aug}}(\cdot|x,a)}[Q(x', a^*(x'))]. \tag{20}$$

Then $\mathcal{T}_c$ maps bounded action-value functions to bounded action-value functions and is a contraction under the sup norm, with modulus at most γ. Consequently, it has a unique fixed point Q_c^*, and the exact iterates satisfy

$$\|\mathcal{T}_c^n Q - Q_c^*\|_\infty \le \gamma^n \|Q - Q_c^*\|_\infty.$$

More generally, an approximate update with uniform operator error at most ε satisfies

$$\|Q_n - Q_c^*\|_\infty \le \gamma^n \|Q_0 - Q_c^*\|_\infty + \frac{\varepsilon(1 - \gamma^n)}{1 - \gamma}.$$

The delay weight plays no role in the modulus: with the kernel, the contrast, and the selection rule fixed, the $\bar{\gamma}(h)[\Delta C]_+$ term is additive and cancels in the difference, leaving the standard γ-contraction of the bootstrap. Stated plainly, this is the ordinary one-step contraction, not a new modulus from delay modeling. The result is therefore a fixed-kernel critic guarantee, not a convergence or performance guarantee for the learned neural implementation.

Taking the supremum over (x, a) proves the result (full steps in Appendix E.2). The conclusion covers the fixed-kernel tabular map only – not the learned neural update, the history encoder, or the learned delay distribution – and the delay weight enters only the consequence signal (Eq. (8)), not the reward target (Eq. (7)). The weight itself is bounded ($\gamma^K \le \bar{\gamma}(h) \le 1$, Appendix E.2). The per-episode update order is listed in Appendix I.

6 Experimental Protocol

6.1 Controlled delayed-cost benchmark

The primary benchmark is a six-dimensional continuous-state, five-action controlled CMDP emitting trajectories $\{(s_t, a_t, r_t, c_t)\}_{t=0}^{T-1}$, where c_t comes from a delayed consequence pipeline. The experimental suite contains standard, noisy, shifted, randomized, deceptive-reward, and resource-collapse variants, with held-out adversarial environments for transfer. The environments expose consequence metadata needed for controlled diagnostics; this same availability limits what can be inferred about real-world causal discovery. Seeds are fixed before training and reused across methods, with no selection on evaluation returns; CSR is episode-level before averaging, and tables lacking replicate files carry no CIs or superiority claims.

For each environment, training and evaluation use separate resets; evaluation freezes the agent and records episode-level return, discounted consequence, and constraint satisfaction. Reference aggregates average over environment variants and three training seeds without pooling episodes as independent replicates (CSR is episode-level and one seed spans variants). The reference CSR is the percentage of evaluation episodes within budget.

The reference stress test is the primary quantitative protocol for the headline adversarial and transfer claims. We do not pool exploratory runs with different training configurations into these results, since delayed-credit methods can be sensitive to training budget and evaluation protocol.

6.2 Baselines and ablations

The reference comparison includes PPO, A2C, DQN, CPO-FO, RCPO, PID-Lag, and SAC-Lag. CPO-FO is the first-order baseline, not an exact natural-gradient CPO. No significance value is reported without retained replicate-level records: each ablation keeps the common training budget and changes the named direction only, separating mechanism from capacity effects.

The baseline set spans unconstrained policy optimization, value-based learning, trust-region constrained optimization, and Lagrangian methods (Achiam et al., 2017; Stooke et al., 2020; Liu et al., 2022). Two standard delayed-credit answers are already inside this set: CPO-FO and RCPO train cost critics with GAE (TD(λ) at λ=0.95), and every CCPL-family policy is recurrent (GRU backbone, delay model on its hidden state). A dedicated cost-decomposition method (HCA/RUDDER on constraint cost), a known-delay augmentation control, and a tuned OmniSafe suite are not evaluated here; the reported numbers (Appendix Table 6) are from the controlled benchmark suite; Section 7.2 reports the Safety-Gymnasium comparison with its per-seed caveat.

We use the reward–constraint trade-off as the primary endpoint: we report J_r, J_c, and CSR together, keep transfer separate from in-distribution results, and never treat diagnostic endpoints as policy-performance metrics. See Appendix I for the qualitative SafeRoute sanity check.

7 Results: Attribution, Conditional Safety, and External Validation

7.1 Reference stress test and mechanism boundary

The primary stress test compares CCPL with standard constrained baselines and the no-attribution ablation under the same reference protocol. It tests whether attribution improves the safety signal when consequences are delayed; it is not intended as a comparison of every downstream policy metric or training budget.

The practical significance is conditional but concrete: when costs are immediate, standard constrained RL may be sufficient; when delayed costs must be traced, audited, or used to debug a controller, CCPL adds a measurable provenance signal and an uncertainty-aware fallback. The experiments therefore ask not whether attribution wins everywhere, but when better attribution changes safety under delayed feedback.

Table 6 reports the benchmark summary; configuration details are listed in Appendix 5.

The reference stress test provides our main safety signal. In the three held-out adversarial environments used by that protocol, CCPL reached 100% CSR, compared with 60–65% for SAC-Lag and the no-attribution CCPL ablation. The corresponding zero-shot transfer check showed an 18.3 percentage-point CSR drop for CCPL versus 26.2 points for the causal-attribution ablation. These are reference short-budget results under the stated evaluation protocol; broader training-budget and deployment effects remain open.

7.2 External simulator checks and adversarial transfer

Because the primary suite is controlled, we also evaluate external simulator tasks. On two Safety-Gymnasium tasks (Ray et al., 2019; Ji et al., 2023), the wrapper inserts a fixed 5-step delay and discretizes continuous actions; Appendix B.2 reports CCPL CSR 94% on SafetyPointGoal1 and 87% on SafetyPointGoal2 in the five-seed run. We use this external check for protocol validation and report the reward trade-off directly: CPO-FO and PPO obtain higher mean reward on both tasks, while CSR differences are small. The controlled six-dimensional CMDP remains the setting for SCM-based causal validation.

Table 1: Corrected Shapley ablation under label-free hindsight. Means across five seeds; the target is cost-weighted simulator provenance.

Profile	Hindsight		No-split	
	Overlap ↑	TAE ↓	Overlap ↑	TAE ↓
fixed5	0.169	3.459	0.167	2.938
geometric	0.197	3.114	0.078	3.892
heavytail	0.190	4.208	0.072	4.579

CCPL's mean inference time ($\sim$1.1–1.5 ms) is roughly five to six times PPO's and SAC-Lag's, consistent with its extra ICN forward pass. We therefore interpret the Safety-Gymnasium numbers as protocol evidence and leave broader deployment performance for future evaluation.

7.3 Attribution and delay diagnostics

The ICN tracks the interventional simulator contrast (E3: MAE 0.0255 ± 0.0067, $r = 0.971 \pm 0.012$, sign agreement 0.953 ± 0.016 at episode 1000); these metrics decline modestly after about episode 1300. The delay model recovers a known constant delay only after longer training (E4: mean 9.9992 vs. 10 at 2000 episodes, approximately 5.26 at 100). Full trajectories and checkpoint results are in Appendix B.

7.4 Label-free attribution and uncertainty controls

The newer attribution protocol withholds source labels from the agent, scoring the cost-weighted provenance estimand (Appendix F) on fixed5, geometric, and heavytail delay profiles (five seeds each). Naive visible-step attribution has zero overlap in every profile, whereas hindsight reaches 0.169–0.197 overlap and 3.11–4.21 steps of temporal attribution error. The corrected Shapley comparison is: For fixed5, we use the paired-seed comparison. Across 15 paired profile–seed comparisons, no-split minus hindsight is -0.079 overlap and $+0.209$ steps TAE: the split improves the pooled distribution, while the fixed-delay effect is inconclusive. A soft-posterior control scores $0.091/0.079$ overlap on bimodal/fixed5 against $0.095/0.093$ for the full Shapley path: most of the gain over argmax comes from keeping the posterior soft, not from the split itself.

Two training-free baselines put these numbers in context. A uniform split over the candidate window scores $0.48/0.52$ overlap – above every learned method – because ground-truth responsibility is diffuse ($\sim$11 contributors per event), so spreading beats precision by construction. A delay-prior-only posterior scores $0.21/0.07$, so delay kinetics alone explain little. Overlap rank does not transfer to policy rank: hindsight matches or leads every baseline on CSR ($0.98/0.94$ vs. uniform $0.94/0.90$) despite lower overlap. The assignment-time label-noise sweep shows overlap rising from level 0 to 0.25 then plateauing, with no monotone TAE degradation; we report no robustness claim from it.

Finally, the residual-on versus residual-off control is null across 15 paired seeds (-0.0062 overlap, $+0.0650$ TAE steps, -0.0600 reward, -0.0011 cost) – expected by construction, since the residual is broadcast identically across actions and cancels in the argmax (up to clipping): a logged diagnostic audit signal, not a learning target.

Matched-cost frontier. The CCPL and CPO-FO pilots had no common measured cost range across all seeds, so these runs do not support a matched-cost reward comparison.

Where attribution pays. One regime separates on the cost endpoint: bimodal delay, tight budget, and separable rewards. Across five paired seeds, hindsight minus naive is $+0.14$ CSR and -0.278 cost (95% CI $[-0.474, -0.018]$), while the reward difference is -1.183 with a 95% CI that includes zero. This supports CCPL's value as improved delayed-cost accounting without a required reward advantage; full replication details are in Appendix B.3.

Intervention validation. A `do()`-style frozen-suffix protocol forces HOLD vs. SURGE at turn t and takes the delta on created (never observed) cost (n=1200, fixed and stochastic delay). Simulator responsibility shares track the measured delta at r=0.99; exact path-Shapley over the pre-event

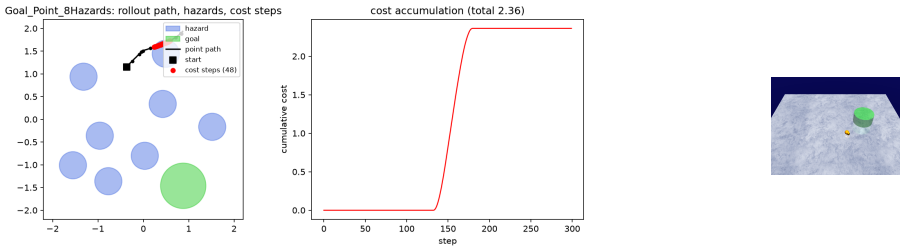


Figure 1: Real `Goal_Point_8Hazards` (GUARD/MuJoCo). Left: top-down rollout path with hazards, goal, and cost accumulation. Right: MuJoCo view near the goal.

window tracks weakly (r=0.43, sign-hit 0.44), consistent with coalition-model sensitivity; naive blame has r=0.00.

Portable patch and shield. A 60-line `CreditPatch` wraps a third-party CPO-FO agent at the `store` level only: across five fixed5 seeds, the uniform prior reaches 0.3265 ± 0.0163 mean overlap and 1.539 ± 0.591 TAE, versus 0.0272 ± 0.0006 overlap and 6.554 ± 0.796 TAE for recency. At this budget, the wrapper changes attribution metrics but leaves reward, cost, and CSR unchanged. A runtime `SafetyPolicy` budget trip reduces mean violation steps from 5.10 to 3.50 across three seeds, a 31% reduction, while mean reward changes from 8.18 to 7.44. The action gate alone is nearly neutral in this run; the budget-aware gate is the condition that reduces violations.

GUARD four-condition comparison. We compare undelayed feedback, delayed feedback, oracle timestamp correction, and the hindsight posterior on `Goal_Point_8Hazards`, both on the local 8-hazard analogue and in GUARD MuJoCo. The analogue at tight budget shows a dose-response: oracle (TAE 0.00) $R3.61/C7.59$, posterior (TAE 2.59) $R3.45/C5.46$/CSR 1.00 (TAE on clean-emission episodes only, 13–17 of 30 per condition). On real physics (10 episodes $\times$ 500 steps $\times$ 3 seeds, fixed delay 5), the posterior reaches TAE 0.43 against the oracle 0.00 with CSR 0.87–1.00; task returns are near zero for all methods at this scale. Posterior cost is 5.46, below the oracle cost of 7.59; our analysis does not explain this difference. Algorithm 3 lists the GUARD, intervention, and patch procedures step by step.

8 Limitations and Discussion

Simulator SCM labels train and validate the action-level contrast in our controlled benchmark; the operational assignment path receives no such labels. The intervention diagnostic therefore establishes agreement with known simulator interventions, while observational causal identification remains untested. Our Shapley scaffold distributes joint contributions but does not define a path-specific estimand for genuinely joint responsibility.

The controlled benchmark is one CMDP family. Safety-Gymnasium and GUARD/MuJoCo checks broaden protocol coverage across simulated environments, but do not establish real-world validity or deployment safety. Other limitations include deterministic history summaries, late or unreliable delay calibration, heuristic uncertainty/OOD margins without coverage guarantees, and multiplier updates without a convergence certificate. We leave dedicated cost- decomposition, known-delay augmentation, and tuned OmniSafe comparisons to future work.

9 Conclusion

CCPL provides an auditable way to assign delayed safety costs to likely source actions while retaining unresolved responsibility as state-level risk. In the bimodal delayed-cost setting, attribution improved constraint satisfaction without a detectable reward change, and responsibility shares tracked simulator intervention effects. This is evidence for a conditional safety benefit, not for general causal validity or deployment performance.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Eitan Altman. *Constrained Markov Decision Processes*. Chapman and Hall/CRC, 1999.

Jose A. Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. RUDDER: Return decomposition for delayed rewards. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18(1): 6070–6120, 2017.

Yinlam Chow, Ofir Nachum, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado van Hasselt, Greg Wayne, Satinder Singh, Doina Precup, and Remi Munos. Hindsight credit assignment. *arXiv preprint arXiv:1912.02503*, 2020. URL https://arxiv.org/abs/1912.02503.

Matthew Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable MDPs. *AAAI Fall Symposium Series*, 2015.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Juntao Dai, and Yaodong Yang. Safety-gymnasium: A unified safe reinforcement learning benchmark. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.nips.cc/paper/2023/hash/3c557a3d6a48cc99444f85e924c66753-Abstract-Datasets_and_Benchmarks.html.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. OmniSafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25, 2024.

Konstantinos V. Katsikopoulos and Sascha E. Engelbrecht. Markov decision processes with delays and asynchronous cost collection. *IEEE Transactions on Automatic Control*, 48(4):568–574, 2003.

Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems*, 30, 2017.

Yongshuai Liu, Jiaxin Ding, and Xin Liu. IPO: Interior-point policy optimization under constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(4):4940–4947, 2020.

Yongshuai Liu, Alon Halev, Krzysztof Chalupka, et al. Constrained variational policy optimization for safe reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162. PMLR, 2022. URL https://proceedings.mlr.press/v162/liu22b.html.

Nikola Milošević, Johannes Müller, and Nico Scherf. Embedding safety into RL: A new take on trust region methods. *Proceedings of the 42nd International Conference on Machine Learning*, 267:44199–44224, 2025. URL https://proceedings.mlr.press/v267/milosevic25a.html.

Washim Uddin Mondal and Vaneet Aggarwal. Reinforcement learning with delayed, composite, and partially anonymous reward. *arXiv preprint arXiv:2305.02527*, 2023. URL https://arxiv.org/abs/2305.02527.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free RL can be a strong baseline for many POMDPs. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, 2022.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. In *OpenAI technical report*, 2019. URL https://openai.com/index/benchmarking-safe-exploration-in-deep-reinforcement-learning/.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. In *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Lloyd S. Shapley. A value for n-person games. In *Contributions to the Theory of Games II*, pp. 307–317. Princeton University Press, 1953. doi: 10.1515/9781400881970-018.

Aivar Sootla, Alexander I. Cowen-Rivers, Taher Jafferjee, Ziyan Wang, David Mguni, Jun Wang, and Haitham Ammar. Saute RL: Almost surely safe reinforcement learning using state augmentation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, 2022.

Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 9133–9143. PMLR, 2020. URL https://proceedings.mlr.press/v119/stooke20a.html.

Chen Tessler, Daniel J. Mankowitz, and Shie Mannor. Reward constrained policy optimization. *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1805.11074.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Bo Yuan, Xueqian Wang, and Dacheng Tao. Penalized proximal policy optimization for safe reinforcement learning. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, 2022. URL https://arxiv.org/abs/2205.11814.

Yiming Zhang, Quan Vuong, and Keith W. Ross. First order constrained optimization in policy space. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Yudi Zhang, Yali Du, Biwei Huang, Ziyan Wang, Jun Wang, Meng Fang, and Mykola Pechenizkiy. Interpretable reward redistribution in reinforcement learning: A causal approach. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.18427.

A Notation

Table 2 collects the symbols used throughout this manuscript for reference.

B Supplementary Figures

B.1 E3/E4 diagnostic detail

The five-seed E3 run measures ICN agreement with its SCM training labels at checkpoints every 50 episodes through episode 2000. We report episode 1000 because it lies within the observed plateau across all three metrics. All three metrics decline modestly after about episode 1300 (Table 4, Figure 7); possible explanations include ICN overfitting, drift in h_t, and optimizer instability.

Table 2: Notation summary.

Symbol	Meaning
s_t, a_t, r_t, c_t	state, action, reward, consequence (cost) at step t
x_t, z_t	physical state, pending-consequence state
ξ_t, η_t	exogenous disturbances driving z_{t+1} and c_t
h_t	finite history representation used in place of z_t
$\gamma \in (0, 1)$	base discount factor
T, d	horizon, constraint budget
$\tau_t \in \{0, \dots, K\}$	consequence delay associated with action a_t
$p_\varphi(k \mid h_t)$	learned delay distribution
$\bar{\gamma}(h_t)$	effective delay weight, Eq. (5)
a_0	reference (baseline) action used for the causal contrast
$C^{\mathrm{do}(a)}(s, h)$	interventional consequence under $\mathrm{do}(a)$
$\Delta C(s, a, h)$	true marginal causal contrast, Eq. (9)
$f_{\mathrm{total},\psi}$, $f_{\mathrm{base},\psi}$	ICN total / baseline heads, parameters ψ
$\widehat{\Delta C}_\psi$	ICN's learned contrast estimate
σ_ψ	ICN uncertainty (scale) head output
Q_r, Q_c	reward / consequence critics
Q_r^-, Q_c^-	target-network copies of Q_r, Q_c
y_t^r, y_t^c	reward / consequence critic regression targets (main agent; both λ-free)
$\tilde{y}_t^c$	single-Q helper target with λ folded in (ablation path only)
d_t	done flag (distinct from budget d)
$\lambda_\phi(s)$, $\lambda_{\max}$	state-conditioned multiplier, its upper bound
Π_{ICN}, Π_{Q_c}	clipped penalty terms used at action selection
$\Pi_{\mathrm{action},t}$, $\Pi_{\mathrm{state},t}$	action penalty / unresolved state residual, Eq. (17)
w_i	per-sample supervision weight (zero when a label is invalid)
$w_{t,\tau}$	posterior responsibility of lag τ at time t
$\mathcal{T}_t, m$	candidate lag set, its size
$H_t, \widetilde{H}_t, \eta$	posterior entropy, normalized entropy, gate threshold
ℓ	Laplace likelihood, Eq. (13)
$x_{t,\tau}, v_t, \phi_{t,\tau}$	Shapley contributions, coalition game, Shapley values
$x = (s, h)$, P_{aug}	augmented state, fixed augmented kernel
$K, k_{\min}$	maximum delay, minimum positive delay (when assumed)
$w_{\max}, \tau^\star$	maximum posterior weight, its lag
$\mathcal{T}_c$	fixed-kernel consequence-critic Bellman operator
$J_r(\pi), J_c(\pi)$	expected discounted return / consequence under π
CSR	constraint-satisfaction rate (episode-level indicator, Eq. (38))
Overlap, TAE	cost-weighted provenance agreement / temporal error, Eq. (40)

The E4 diagnostic tracks whether the learned delay distribution $p_\varphi(\tau \mid h)$ recovers a known constant delay. Two runs under the same nominal protocol differ substantially: with an observed constant delay of 10 and training limited to 100 episodes, the predicted delay mean was approximately 5.26 (calibration MAE approximately 4.742, mass near $\tau = 5$ with a long low-mass tail; Figure 6, left), while at 2000 training episodes the mean was 9.9992 (MAE 0.0008, mass at $\tau = 10$; Figure 6, right). The runs differ in training duration (100 versus 2000 episodes). Because we did not evaluate intermediate checkpoints, the comparison does not establish a monotonic calibration trend.

B.2 External-validation detail

The figures illustrate the implementation data flow, delayed-consequence alignment, the ICN's two-head factorization, and the observational versus interventional contrast defined in the main text.

B.3 Bimodal-delay replication

We independently repeated the bimodal-delay, tight-budget protocol with five paired seeds, 100 training episodes, and a 20-episode evaluation window. The reward vector was $(0.05, 0.85, 0.25, 0.70)$, the budget was 2.5, and delay labels were withheld. Relative to visible-step blame, learned hindsight reduced mean episodic consequence from 2.172 to 1.894; the paired difference was -0.278 with 95% bootstrap CI $[-0.474, -0.018]$. Mean CSR changed from 0.84 to 0.98, while the reward difference

Table 3: Safety-Gymnasium task comparison, mean across five seeds. J_c is mean episodic consequence; CSR uses Eq. (38).

	SafetyPointGoal1			SafetyPointGoal2		
Method	Rew.	$J_c \downarrow$	CSR↑	Rew.	$J_c \downarrow$	CSR↑
CCPL	1.697	4.843	94.0	1.119	8.210	87.0
PPO	5.469	5.064	99.0	5.378	10.952	90.0
CPO-FO	3.690	3.634	97.0	2.982	8.576	91.0
SAC-Lag	-0.125	1.827	98.0	-0.463	5.249	89.0

was not statistically distinguishable from zero. Attribution overlap increased from 0 to 0.095 and TAE decreased from 10.020 to 5.953, reproducing the conditional safety result.

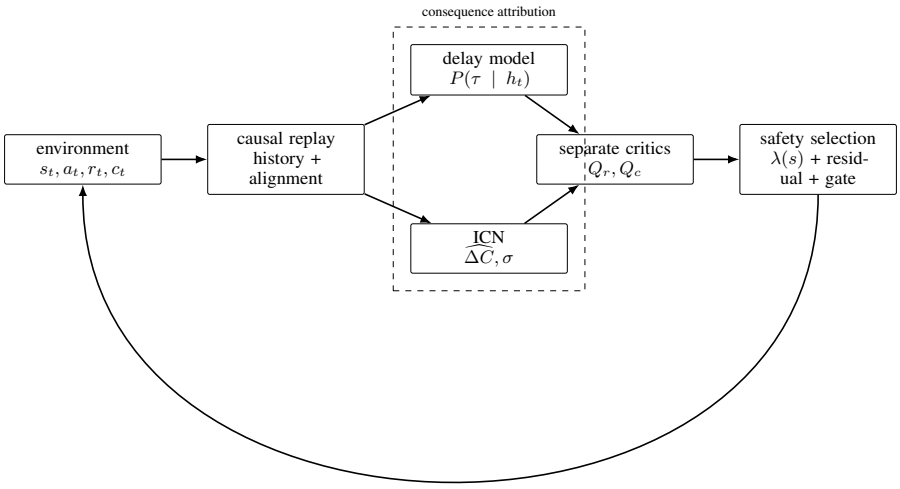


Figure 2: Implementation-level CCPL data flow. Delayed feedback is retained in replay, aligned when a source delay is available, and used by the delay model, ICN, critics, and safety-selection gate.

Table 4: E3 checkpoints, 5-seed mean $\pm$ std. Episode 1000 is the reported checkpoint; episode 2000 is the final checkpoint.

Episode	MAE $\downarrow$	Pearson r $\uparrow$	Sign agreement $\uparrow$
0 (untrained)	0.1215 ± 0.0403	0.607 ± 0.177	0.575 ± 0.129
100	0.0476 ± 0.0088	0.912 ± 0.020	0.894 ± 0.029
500	0.0339 ± 0.0113	0.960 ± 0.017	0.944 ± 0.021
1000 (headline)	$\mathbf{0.0255 \pm 0.0067}$	$\mathbf{0.971 \pm 0.012}$	$\mathbf{0.953 \pm 0.016}$
2000 (final)	0.0325 ± 0.0065	0.950 ± 0.026	0.942 ± 0.025

C ALGORITHM

Algorithm 1 gives the full CCPL training loop corresponding to the data flow described informally in Section D and Figure 2. It makes explicit the order of operations summarized in Appendix I: history and replay bookkeeping happen before any network update, and the ICN, delay network, and critics are updated using labels computed *before* the current step's target-network soft update, so that no update within a single step depends on its own output.

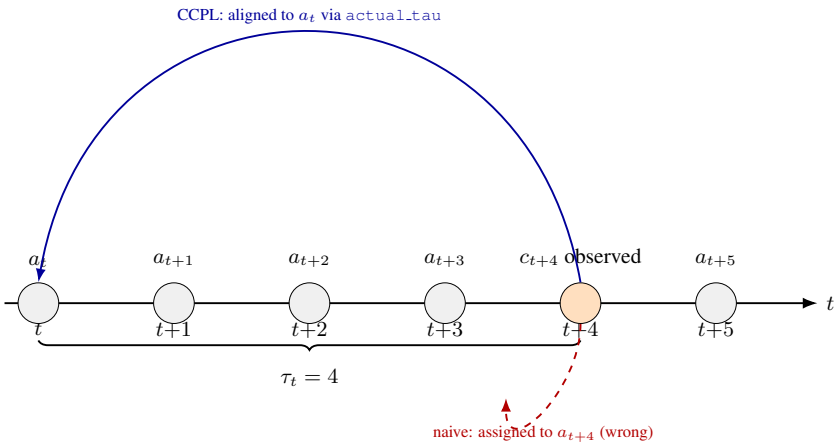


Figure 3: Delayed-consequence alignment. Action a_t initiates a process whose cost c_{t+4} is only observed four steps later, after $a_{t+1}, \ldots, a_{t+3}$ have already executed. A naive learner assigns the observed cost to whichever action is active at emission time; CCPL instead walks the replay buffer back by the recorded delay (`actual_tau` $= 4$ in this example) and writes the aligned consequence onto the transition for a_t. The alignment shown here requires the environment to expose `actual_tau`; Section 8 discusses the fallback behavior when it is unavailable.

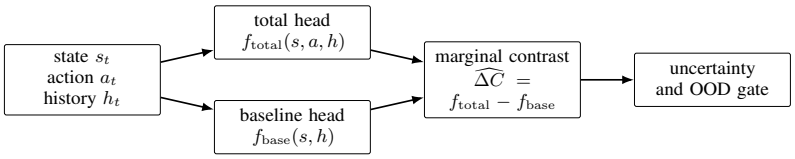


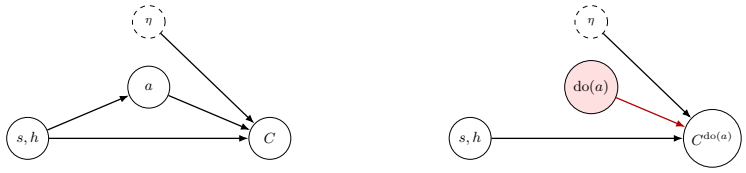
Figure 4: ICN factorization used in the implementation. The baseline retains the state and history but omits the current action; this is an action-marginal baseline, not a state-free baseline. The difference is trained only where a valid SCM or aligned consequence label exists.

Two aspects of Algorithm 1 are easy to miss when reading the per-mechanism sections in isolation and are worth stating together here. First, the multiplier λ_ϕ never appears inside either critic target (y_i^r or y_i^c); it enters only at the action-selection step and in its own update rule. This is what keeps the critic regression targets stationary as λ_ϕ moves, which is the practical justification (not a theorem) for treating the multiplier update and the critic updates as only weakly coupled. Second, the ICN update happens *before* the critic update within the same batch, so y_i^c always uses the current step's ICN output rather than a stale one; the delay network update happens between the two, so $\bar{\gamma}(h_i)$ used in y_i^c also reflects the same step's information. Neither ordering choice is claimed to be provably optimal; we report it because Theorem 1 analyzes a fixed-kernel operator that assumes $\bar{\gamma}$ and ΔC are held fixed within an update, and this is the concrete approximation to that assumption that the implementation makes.

D Complete Update and Data Flow

For each transition, the replay record contains the current state s_t, action a_t, reward r_t, emitted consequence c_t, next state s_{t+1}, done flag, current and next hidden states, a fixed-length history, aligned consequence, and validity flags. The aligned flag is set only when the environment identifies a valid source delay. The SCM flag identifies records for which simulator labels can be generated.

The online sequence is: reset hidden state and causal history; compute all-action ICN contrasts and uncertainty; compute $\bar{\gamma}(h_t)$ and $\lambda_\phi(s_t)$; score and execute an action; push the transition; align the



(a) Observational SCM: a depends on s; a naive contrast confounds the two.

(b) Interventional SCM: incoming edges into a are cut, isolating a's effect.

Figure 5: Observational versus interventional queries for the same structural causal model. (a) In the observational graph, the action a has its own incoming edge from (s, h) (the policy that chose it), so a plain conditional contrast $\mathbb{E}[C \mid s, a, h] - \mathbb{E}[C \mid s, a_0, h]$ mixes the causal effect of a with whatever made the policy more likely to choose a in that state. (b) $\mathrm{do}(a)$ severs that incoming edge and fixes a by external intervention, so $C^{\mathrm{do}(a)}(s, h)$ isolates a's effect. The ICN contrast $\widehat{\Delta C}_\psi$ (Eq. (9)) is trained only on labels generated by simulating (b); it is therefore an interventional estimand evaluated in this simulator (Section 8).



Figure 6: Learned delay distribution $P(\tau \mid h)$ against a true constant delay of $\tau = 10$. Left: 100 training episodes, predicted mean 5.26 (MAE 4.742). Right: 2000 training episodes, predicted mean 9.9992 (MAE 0.0008). Both runs are reported; training duration is a plausible explanation for the gap.

emitted consequence when `actual_tau` is valid; sample prioritized replay; and update the ICN, delay network, critics, multiplier, priorities, and target networks in that order.

The ICN pretraining uses two stages. The baseline head is fitted to same-state baseline targets, then the total head is fitted to per-action total consequences while the uncertainty head is trained jointly. Pretraining diagnostics report held-out agreement with simulator labels.

E Derivations and Assumptions

E.1 Effective delay weight

Since $0 < \gamma < 1$ and $0 \le k \le K$, $\gamma^k \in [\gamma^K, 1]$. Because $p(k \mid h)$ is non-negative and sums to one,

$$\gamma^K \le \bar{\gamma}(h) \le 1. \tag{21}$$

If a known support condition gives $p(k \mid h) = 0$ for $k < k_{\min}$, with $k_{\min} \ge 1$, then $\bar{\gamma}(h) \le \gamma^{k_{\min}} < 1$. This does not say that arbitrary unknown stochastic delays satisfy the support condition.

For an estimated distribution $\widehat{p}(\cdot \mid h)$, the weighting error obeys

$$\left|\widehat{\bar{\gamma}}(h) - \bar{\gamma}(h)\right| = \left|\sum_{k=0}^{K} (\widehat{p}(k \mid h) - p(k \mid h))\gamma^k\right| \tag{22}$$

$$\le \sum_{k=0}^{K} |\widehat{p}(k \mid h) - p(k \mid h)|. \tag{23}$$

An ℓ_1 error bound on the delay distribution therefore transfers to the effective weighting signal; source-action identification and policy outcomes are evaluated separately.

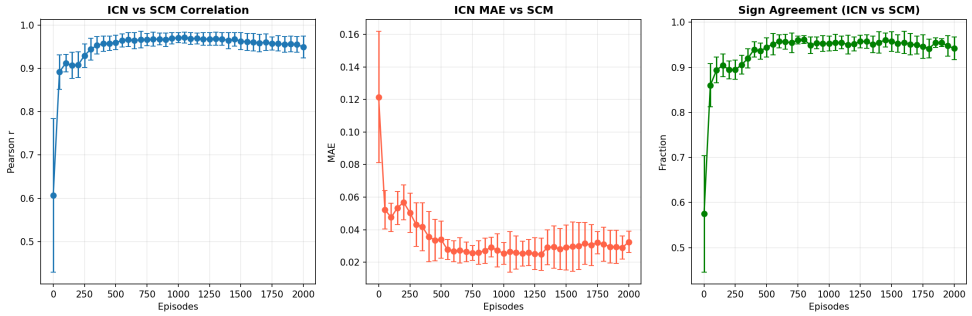


Figure 7: ICN agreement with SCM reference labels over 2000 training episodes, 5-seed mean ± one standard deviation. Correlation (left) and sign agreement (right) rise sharply in the first ∼500 episodes, plateau between roughly episode 750 and 1300, then decline modestly toward episode 2000; MAE (center) follows the reverse trend. The main text reports episode 1000; this figure shows the full trajectory, including the later decline.

E.2 Fixed-kernel contraction calculation

Proposition 1 (Bounds on the learned delay weight)**.** *If $p(k \mid h)$ is a probability distribution on $\{0, \dots, K\}$, then $\gamma^K \le \bar{\gamma}(h) \le 1$. If there is a uniform positive minimum delay $k_{\min} \ge 1$, then $\bar{\gamma}(h) \le \gamma^{k_{\min}} < 1$.*

Proof. The values γ^k lie in $[\gamma^K, 1]$ and $\bar{\gamma}(h)$ is their convex combination. Under the support restriction, every nonzero term is at most $\gamma^{k_{\min}}$. □

For fixed augmented kernel P_{aug}, fixed p_φ, bounded ΔC, and a fixed deterministic action-selection rule $a^*(x')$, the tabular update matching the consequence-critic target is

$$(\mathcal{T}_c Q)(x, a) = \bar{\gamma}(h)[\Delta C(x, a)]_+ + \gamma \mathbb{E}_{x' \sim P_{\text{aug}}(\cdot \mid x, a)}[Q(x', a^*(x'))]. \tag{24}$$

Two target helpers exist and must not be confused. The main agent keeps both critics free of the multiplier: the reward target is $r_t + \gamma(1 - d_t)Q_r^-$ and the consequence target is $\gamma_e \cdot \text{attributable}_t + \gamma(1-d_t)Q_c^-$ with a^* held fixed for the analysis. This is the form for which the stationarity discussion (multiplier outside both targets) holds. The `td_target` helper instead folds $-\lambda \cdot \text{scale} \cdot \gamma_e[\Delta C]_+$ into a single reward-like target; it belongs to the single-Q ablation path (where the multiplier is deliberately baked in) and is not used by the main agent. For the fixed-selection-rule operator, the consequence term is additive and fixed, so with a^* held fixed, $|(\mathcal{T}_c Q) - (\mathcal{T}_c Q')| \le \gamma \|Q - Q'\|_\infty$ directly; the delay weight contributes nothing to the modulus, which is the ordinary one-step γ. For any bounded functions Q, Q', the additive consequence terms cancel: with a^* fixed, $|Q(x', a^*(x')) - Q'(x', a^*(x'))| \le \|Q - Q'\|_\infty$ pointwise, so

$$|(\mathcal{T}_c Q) - (\mathcal{T}_c Q')| \le \gamma \mathbb{E}_{x' \sim P_{\text{aug}}}[\|Q - Q'\|_\infty] \tag{25}$$

$$\le \gamma \|Q - Q'\|_\infty. \tag{26}$$

Thus the fixed-kernel tabular map has modulus at most $\gamma < 1$. Banach's fixed-point theorem therefore gives a unique fixed point for this fixed-kernel map and, under exact iteration,

$$\|Q_n - Q^*\|_\infty \le \gamma^n \|Q_0 - Q^*\|_\infty. \tag{27}$$

If each approximate update has uniform error at most ε, the same contraction argument gives

$$\|Q_n - Q^*\|_\infty \le \gamma^n \|Q_0 - Q^*\|_\infty + \frac{\varepsilon(1 - \gamma^n)}{1 - \gamma}. \tag{28}$$

This calculation concerns the fixed tabular operator. It does not cover neural function approximation, changing delay kernels, target-network lag, replay sampling, or online SCM estimation. The reward critic uses the ordinary one-step target; the delay weight is applied to the attributed consequence signal rather than to an unobserved multi-step reward bootstrap.

Algorithm 1: CCPL training loop

Data: environment Env, simulator SCM oracle (training only), replay buffer $\mathcal{B}$
Initialize: $\psi, \phi, \theta_r, \theta_c$ (ICN, multiplier, reward critic, consequence critic); target copies $\theta_r^- \leftarrow \theta_r$, $\theta_c^- \leftarrow \theta_c$; delay network p_φ; empty causal history h_0

for *episode* $= 1, 2, \ldots$ **do**
  reset Env, hidden state, and causal history $h \leftarrow h_0$;
  for $t = 0, 1, \ldots, T-1$ **do**
    compute $\widehat{\Delta C}_\psi(s_t, a, h_t)$ and $\sigma_\psi(s_t, a, h_t)$ for all $a \in \mathcal{A}$;
    compute $\bar{\gamma}(h_t)$ from $p_\varphi(\cdot \mid h_t)$ (Eq. (5));
    compute $\lambda_\phi(s_t) = \lambda_{\max} \operatorname{sigmoid}(f_\phi(s_t))$;
    `// action selection; penalty terms constructed in Algorithm 2`
    $a_t \leftarrow \arg\max_a \{Q_r(s_t, a) - \Pi_{\mathrm{ICN}}(s_t, a, h_t) - \lambda_\phi(s_t)\Pi_{Q_c}(s_t, a)\}$;
    apply HallucinationGate: widen penalty by a bounded margin when σ_ψ or an OOD score exceeds threshold;
    execute a_t, observe $(s_{t+1}, r_t, c_t, \mathrm{done}_t)$;
    push $(s_t, a_t, r_t, c_t, s_{t+1}, \mathrm{done}_t, h_t, h_{t+1})$ to $\mathcal{B}$, with SCM and alignment validity flags;
    if *oracle/diagnostic mode exposes* `actual_tau` *for this consequence* **then**
      walk back `actual_tau` steps in $\mathcal{B}$ and call `set_aligned_consequence` on the source transition;
    else if *hindsight mode and* $c_t > 0$ **then**
      run Algorithm 2 on the candidate window; accumulate soft credit and delay targets;
    update h_{t+1} from h_t, s_t, a_t;
  if $|\mathcal{B}| \geq$ *batch size* **then**
    sample prioritized batch $\{(s_i, a_i, r_i, c_i, s_i', h_i, w_i)\}_{i=1}^B \sim \mathcal{B}$;
    `// 1. ICN update on valid labels; hindsight uses the posterior`
    update ψ by descending $\mathcal{L}_{\mathrm{ICN}}(\psi)$ and the Laplace uncertainty loss on $\{i : w_i > 0\}$;
    in the controlled causal-validation configuration only, periodically recalibrate ICN heads against fresh SCM labels (held out from the training batch); the label-free operational configuration omits this step;
    `// 2. delay network update`
    update φ by descending $\mathcal{L}_{\mathrm{delay}}(\varphi)$;
    `// 3. critic targets, computed with target networks, current` $\bar{\gamma}$, `and current ICN`
    $y_i^r \leftarrow r_i + \gamma(1 - d_i) Q_r^-(s_i', a_i')$ for $a_i' \leftarrow \arg\max_a Q_r^-(s_i', a)$;
    $y_i^c \leftarrow \bar{\gamma}(h_i)[\widehat{\Delta C}_\psi(s_i, a_i, h_i)]_+ + \gamma(1 - d_i) Q_c^-(s_i', a_i')$;
    `// consequence-only idealization; implemented target in text`
    `// 4. critic update`
    update θ_r, θ_c by descending weighted TD error against y_i^r, y_i^c;
    `// 5. multiplier update`
    update ϕ toward the target computed from global episode violation, local attributed consequence, and recent hit frequency, with warm-up scaling;
    `// 6. priorities and target networks`
    update replay priorities from TD error magnitude;
    soft-update $\theta_r^- \leftarrow \rho\theta_r^- + (1 - \rho)\theta_r$, and likewise for θ_c^-;

E.3 Weighted ICN risk and the learned contrast

Let $X = (S, A, H)$, let $Y = \Delta C$, and let $W \geq 0$ be a label weight with $\mathbb{E}[W \mid X] > 0$. The population weighted square risk is

$$R(f) = \mathbb{E}\left[W(Y - f(X))^2\right]. \tag{29}$$

Algorithm 2: Hindsight credit assignment for one observed cost

Data: observed cost $c_t > 0$, candidate window $\mathcal{T}_t$ (recent lags), delay distribution $p_\varphi(\cdot \mid h_t)$, ICN outputs $(\widehat{\Delta C}_{t,\tau}, \sigma_{t,\tau})$, temperature T, scale floor $\sigma_{\min}$, joint margin, entropy gate η

form prior $p_\tau \propto \max(p_\varphi(\tau \mid h_t), 0)$, renormalized;
clamp scales $s_\tau \leftarrow \max(\sigma_{t,\tau}, \sigma_{\min})$;
if *oracle mode with a usable lag* $\tau^\circ \in \mathcal{T}_t$ **then**
    return point mass on τ° labeled `oracle`;
if *oracle mode without a usable lag* **then**
    return zero weights labeled `oracle-missing` (writes nothing);
compute Laplace log-likelihoods $\log \ell_\tau = -|c_t - \mu_\tau|/(s_\tau T) - \log s_\tau$;
$w_\tau \propto p_\tau \ell_\tau$ normalized (uniform fallback if degenerate);
if $c_t - \max_\tau[\widehat{\Delta C}_{t,\tau}]_+ > \mathit{margin} \cdot s_{\tau^\star}$ *and total predicted signal is non-negligible* **then**
    mark joint; split by Shapley values of $v_t(S) = \min\{c_t, \sum_{\tau \in S} p_\tau[\widehat{\Delta C}_{t,\tau}]_+\}$ (exact ≤ 12 players, sampled beyond, top-8 truncation, prior-only fallback);
compute $w_{\max}$, $\tau^\star$, H_t, $\widetilde{H}_t$ from the *pre-split* posterior;
$\Pi_{\text{action},t} \leftarrow w_{\max}[\widehat{\Delta C}_{t,\tau^\star}]_+$;
$\Pi_{\text{state},t} \leftarrow \mathbf{1}\{\widetilde{H}_t \geq \eta\}(1 - w_{\max})c_t$;
return final weights (sum to one, credited costs sum to c_t), penalties, and one row per candidate (h, τ, w) as the delay target;

Algorithm 3: Evaluation protocols: GUARD comparison, intervention check, patch demo

Data: task, seeds, budgets; modes $\in$ {undelayed, delayed, oracle, posterior}

for *each (task, seed, policy, mode)* **do**
    train with the mode's cost channel (immediate / delayed scalar / oracle-aligned / posterior-aligned), matched budgets;
    record return, delayed-visible cost, undelayed violation steps, max step cost, and credit-log/observation provenance;
report paired seed-mean differences with bootstrap CIs;
`// do() intervention check`
for *each (seed, forced turn t)* **do**
    replay fixed prefix; force HOLD vs. SURGE at t; freeze suffix;
    delta $\leftarrow$ created-cost(CF) $-$ created-cost(factual);
    correlate *delta* with responsibility shares (control), windowed exact Shapley, and naive blame;
`// third-party patch demo`
wrap baseline `store` only: move observed cost off the current row, scatter backward by posterior weights, keep dual target exact;
compare patched vs. unpatched on overlap/TAE and cost/CSR, paired seeds;

Conditioning on $X = x$ and expanding the square gives

$$R_x(u) = \mathbb{E}[WY^2 \mid x] - 2u\mathbb{E}[WY \mid x] + u^2\mathbb{E}[W \mid x]. \tag{30}$$

The unique conditional minimizer is therefore

$$f^*(x) = \frac{\mathbb{E}[WY \mid X = x]}{\mathbb{E}[W \mid X = x]}. \tag{31}$$

If W is conditionally independent of Y given X, this reduces to $f^*(x) = \mathbb{E}[Y \mid X = x]$. This is a population target for the ICN loss, not a finite-sample or neural-optimization guarantee.

For the two-head factorization, the total head receives (s, a, h) and the baseline head receives (s, h). If their labels are respectively $Y_{\text{total}} = C^{\text{do}(a)}$ and $Y_{\text{base}} = C^{\text{do}(a_0)}$, the population difference is

$$\mathbb{E}[Y_{\text{total}} \mid s, a, h] - \mathbb{E}[Y_{\text{base}} \mid s, h]. \tag{32}$$

It equals the stated causal contrast only under the simulator's label generating assumptions. No observational-identification claim follows from the architecture.

The implementation fits the two heads separately rather than regressing the contrast directly. With normalized weights W_n, it minimizes three terms: total-head MSE $\sum W_n(C_{\text{total}} - c)^2$, baseline-head Huber loss (kink 0.5) against the baseline target, and the Laplace location/scale terms of Section E.5. The total head therefore receives both the MSE gradient and the Laplace location gradient through the same pre-activation; the population analysis above describes the fixed point each head targets.

E.4 Bias under a misspecified or noisy SCM

The identification result above assumes the label $Y = \Delta C$ used at training time is the true interventional contrast. In practice the label is generated by a simulator SCM that may itself be noisy or misspecified relative to the environment's true generative process. We give a first-order account of the resulting bias for the reference diagnostics.

Suppose the label actually observed at training time is $\widetilde{Y} = Y + \varepsilon$, where ε is a label-noise term with $\mathbb{E}[\varepsilon \mid X] = b(X)$ (a possibly state-dependent bias) and $\mathrm{Var}(\varepsilon \mid X) = v(X) < \infty$. Repeating the conditional minimization in the previous subsection with $\widetilde{Y}$ in place of Y gives, when W is conditionally independent of (Y, ε) given X,

$$\widetilde{f}^*(x) = \mathbb{E}[\widetilde{Y} \mid X = x] = f^*(x) + b(x). \tag{33}$$

That is, at the population level and under conditionally independent weights, label noise with zero conditional mean ($b \equiv 0$) does not bias the learned contrast, only its variance; the ICN's minimizer remains f^*. A *systematic* misspecification, however, appears directly as an additive bias term $b(x)$ that the ICN cannot distinguish from a true causal effect, since both are captured by the same conditional expectation. This has two consequences worth stating explicitly. First, the E3 diagnostic (Section 7) measures agreement with the *same* SCM used to generate training labels, so a systematic bias in that SCM relative to the environment's true structural equations would not be detected by E3's agreement metric; E3 upper-bounds only the ICN's ability to fit its own supervision, not the supervision's own correctness. Second, if the bias term $b(x)$ is roughly constant across actions at a fixed state (a "shared offset" misspecification), it cancels in the marginal contrast $\widehat{\Delta C} = f_{\text{total}} - f_{\text{base}}$ used at action-selection time, if both heads inherit the same offset; the contrast is then less affected by that shared state-level bias. It is *not* robust to an action-dependent bias $b(s, a, h) - b(s, a_0, h) \neq 0$, which is exactly the harder case a genuinely misspecified structural model would produce. The assignment-time $\Delta C/\sigma$ distortion sweep reported in Section 7.4 perturbs labels at assignment time; it does not test training-time SCM misspecification, since its training labels remain generated by the benchmark SCM.

E.5 ICN objective and uncertainty

The ICN total head uses (s, a, h) while the baseline head uses (s, h). For valid weights, the normalized objective is well-defined when $\sum_i w_i > 0$. Samples with invalid SCM and alignment masks receive zero causal-supervision weight. The uncertainty head produces $\sigma_i > 0$ through a softplus map and uses

$$\ell_i^{\text{Laplace}} = \log(\sigma_i + \varepsilon) + \frac{|C_i - \widehat{C}_i|}{\sigma_i + \varepsilon}. \tag{34}$$

The gate clips positive consequence estimates and adds a bounded uncertainty margin. These operations are engineering safeguards, not coverage guarantees. The effective scale floor is 10^{-4} (softplus offset) plus 10^{-6} in the loss, and σ shares the total trunk. The baseline Huber kink is part of the implementation, not the population analysis.

E.6 Posterior, Shapley, and residual implementation notes

The Laplace likelihood is evaluated at temperature $T = 1$ with a σ floor of 10^{-2}; the prior is renormalized, and an empty or degenerate posterior falls back to uniform weights. Non-positive costs receive zero weights (there is nothing to attribute), and an oracle mode without a usable lag returns an

explicitly labeled missing-label assignment rather than a silent uniform split. A joint event is declared when no single candidate explains the cost,

$$c_t - \max_{\tau} \widehat{\Delta C}_{t,\tau} > \mathit{margin} \cdot \sigma_{t,\tau^\star}, \qquad \mathit{margin} = 2.0, \tag{35}$$

skipped when the total predicted signal is negligible (untrained ICN must not fake jointness). Exact subset enumeration runs up to 12 players, permutation sampling (128 samples in the agent, 256 in the standalone routine) beyond that, truncated to the top 8 contributors with a prior-only fallback. The final weights are normalized to sum to one, so credited costs sum to the observed c_t: this is conservation of the attribution record, and it deliberately replaces Shapley efficiency ($\sum \phi = v(N)$) at the saturating cap. Source-confidence quantities ($w_{\max}$, action penalty, residual gate) are computed from the pre-split posterior even when the final weights are Shapley values.

F Environment and Metric Details

The six controlled benchmark variants modify consequence noise, state shifts, delay structure, reward/consequence interaction, and resource pressure while retaining the common five-action interface. The adversarial variants are held out for transfer experiments. Main-table results use the stated benchmark protocol and seed count.

F.1 Controlled SCM and hidden-delay protocol

The controlled SCM uses the six-dimensional state $s_t = (r_t, f_t, p_t, u_t, v_t, h_t)$ for resource load, future risk, action priority, system pressure, uncertainty, and hidden penalty. For action a_t, the implementation applies the following deterministic increments to (r_t, f_t, u_t, h_t):

Action	Δr	Δf	Δu	Δh
DEFER	-0.05	$0.08(1-p_t)$	0.05	0.02
PARTIAL	0.10	-0.04	-0.03	-0.01
FULL	0.25	-0.10	-0.06	$0.15 f_t$
INVEST	0.12	-0.20	-0.08	-0.12
REBALANCE	-0.10	0.02	-0.15	-0.05

With independent exogenous noise ξ_t, the next state clips each component to $[0,1]$; p_{t+1} is redrawn uniformly from $[0.1, 0.9]$, and the uncertainty coordinate additionally receives $\mathcal{N}(0, 0.03^2)$. The controlled immediate consequence is

$$\begin{aligned} F_t &= \mathrm{clip}(\Delta r_t f_t + 0.5(h_t + \delta)\mathbf{1}\{a_t = \mathrm{FULL}\}, 0, 1)\,, \\ U_t &= \mathrm{clip}\left(\frac{[r_{t+1} - 0.6]_+}{0.4} + \Delta r_t [r_t - 0.5]_+, 0, 1\right), \end{aligned} \tag{36}$$

$$D_t = \mathrm{clip}(0.6 u_t \mathbf{1}\{a_t = \mathrm{FULL}\} + 0.3 v_t |\Delta u_t|, 0, 1)\,, \qquad c_t^{\mathrm{imm}} = 0.5F_t + 0.3U_t + 0.2D_t, \tag{37}$$

where δ is the configured penalty shift. A delay variable $\tau_t \in \{0, \dots, K\}$ places this consequence in the queue and the environment emits it at $t + \tau_t$. These equations define the SCM used for controlled intervention validation.

The label-free hidden-delay benchmark uses $K = 15$, a 120-step horizon, budget $d = 3.0$, default consequence noise 0.03, and delay distributions $\{5:1\}$ (fixed5), $\{2:0.5, 9:0.5\}$ (bimodal), a normalized geometric profile on delays 1–9, and the normalized heavy-tail profile on $\{1, 2, 3, 8, 11, 14\}$. At assignment time, the operational hindsight path receives neither source-lag nor SCM labels; the explicitly named oracle condition exposes source lags. SCM supervision used to train or validate the ICN is documented separately. Evaluation freezes the policy and computes CSR from the episode total, not from individual steps.

For evaluation episodes $\mathcal{D}$ and budget d,

$$\mathrm{CSR} = \frac{1}{|\mathcal{D}|} \sum_{e \in \mathcal{D}} \mathbf{1}\{J_c^{(e)} \le d\}. \tag{38}$$

This episode-level metric differs from a guarantee that every future trajectory satisfies the constraint.

F.2 DELAYED-COST ATTRIBUTION ESTIMAND

For an observed emission turn t, let $\mathcal{E}_t$ be its clean events. Event e has realized cost C_e, contributor turns J_e, and provenance shares $q_{e,j}$ summing to one over J_e. The evaluator's target is the cost-weighted source distribution

$$q_t(j) = \frac{\sum_{e \in \mathcal{E}_t} C_e q_{e,j}}{\sum_{e \in \mathcal{E}_t} C_e}, \qquad q_{e,j} = 0 \text{ when } j \notin J_e. \tag{39}$$

For predicted source weights $\widehat{q}_t(j)$, overlap and temporal attribution error are

$$\text{Overlap}_t = \sum_j \min(\widehat{q}_t(j), q_t(j)), \qquad \text{TAE}_t = \left| \sum_j j\widehat{q}_t(j) - \sum_j j q_t(j) \right|. \tag{40}$$

These metrics compare assignments with the simulator's provenance record. The entropy-gated state residual is excluded from $\widehat{q}_t$ because it is an action-independent penalty for unresolved risk.

G IMPLEMENTATION AND REPRODUCIBILITY NOTES

Implementation checks covered packaged-layout imports, delay accounting and alignment, hindsight attribution and its ablations, joint-head distillation, intervention controls, the CPO patch, GUARD adapter behavior, offline evaluation, and regression checks. Airspace Guardian, Neurocity, and Safety Audit are simulated demonstrations.

G.1 CONFIGURATION REFERENCE

Table 5 lists the configuration values used in the benchmark reported in Section 7.

Table 5: Configuration reference for the six-variant benchmark reported in Table 6.

Component	Setting	Value / note
Environment	action space	5 discrete actions
	state dimension	6
	max delay K	environment-dependent (see variant config)
	seeds	3, fixed before training
Delay network	architecture	MLP over h_t, categorical output over $\{0, \ldots, K\}$
	loss	weighted categorical cross-entropy, Eq. (6)
	warm-start	bounded online prior when labels are sparse
ICN	heads	total $f_{\text{total}}(s, a, h)$, baseline $f_{\text{base}}(s, h)$, scale $\sigma(s, a, h)$
	training	two-stage pretraining, then joint fine-tuning (App. D)
	uncertainty loss	Laplace objective, Eq. (34)
Critics	architecture	separate Q_r, Q_c, shared encoder not assumed
	target update	soft update
	replay	prioritized, priorities from TD error magnitude
Multiplier	form	$\lambda_\phi(s) = \lambda_{\max} \,\text{sigmoid}(f_\phi(s))$
	$\lambda_{\max}$	bounded
Baselines	PPO, A2C, DQN	stated implementation defaults
	CPO-FO, RCPO, PID-Lag, SAC-Lag	stated implementation defaults
Reporting	seeds per method	3
	evaluation episodes per seed	fixed, frozen policy at evaluation

G.2 SUPPLEMENTARY BENCHMARK AND ABLATION TABLE

Table 6 reports the benchmark and component-ablation values. Left: the full baseline comparison; among methods with 100% CSR, CCPL has higher reward than DQN. Right: component isolation, with each ablation removing one named component.

Table 6: Left: benchmark comparison. Right: component ablation. Both: mean across six controlled benchmark variants and three seeds.

Method	Reward ↑	J_c ↓	CSR (%) ↑
CCPL	4.840	0.120	100.0
PPO	-0.105	0.909	95.5
A2C	1.989	0.913	95.5
DQN	2.515	0.537	100.0
CPO-FO	-0.031	0.610	99.6
RCPO	-0.033	0.612	99.6
PID-Lag	-0.040	0.610	99.6
SAC-Lag	0.430	2.670	93.4

Variant	Reward ↑	J_c ↓	CSR (%) ↑
CCPL	4.840	0.120	100.0
CCPL-NoDelay	2.062	0.666	99.9
CCPL-NoStateλ	3.471	0.284	98.1
CCPL-NoCausal	-1.105	0.538	100.0
CCPL-SingleQ	3.218	0.391	96.7

G.3 Evidence and provenance ledger

Table 7 summarizes each result's protocol, label status, and evidence type; seed-level uncertainty estimates are unavailable for the reference component ablation.

Table 7: Evidence ledger for the reported claims. "Hidden" means that source lags and SCM labels were withheld from the operational attribution path.

Claim	Protocol/source	Label status	Evidence retained
Attribution overlap and TAE	Hindsight benchmark; fixed5, geometric, heavytail	Hidden; five seeds/profile	Per-profile aggregates and paired comparisons
ICN causal agreement	Controlled intervention/SCM diagnostic	SCM labels train the ICN and define validation targets	Measured intervention deltas and simulator contrasts
Reference component ablation	Six-environment benchmark	Privileged SCM labels	Table values; no seed-level CIs
Safety-Gymnasium check	SafetyPointGoal1 and SafetyPointGoal2	Delays inserted	Mean CSR/cost only; no significance claim

G.4 Operational audit pseudocode

For an observed positive consequence c_t, the label-free path is the following finite, auditable computation. Let $\mathcal{T}_t$ be the candidate source window, $p_\varphi(\tau \mid h_t)$ the learned delay prior, and $\ell_\psi(c_t \mid s_{t-\tau}, a_{t-\tau}, h_{t-\tau})$ the ICN likelihood:

$$w_{t,\tau} = \frac{p_\varphi(\tau \mid h_t)\ell_\psi(c_t \mid s_{t-\tau}, a_{t-\tau}, h_{t-\tau})}{\sum_{k\in\mathcal{T}_t} p_\varphi(k \mid h_t)\ell_\psi(c_t \mid s_{t-k}, a_{t-k}, h_{t-k})}, \quad \tau \in \mathcal{T}_t. \tag{41}$$

For the audit ledger (separate from the policy penalties in Eq. (17)), the action-level share is sent to the posterior-weighted source estimate; if $w_{\max}$ is below the confidence threshold, the unresolved remainder is kept as state risk. Thus the logged allocation obeys

$$A_t^{\text{ledger}} + R_t^{\text{ledger}} = c_t, \qquad A_t^{\text{ledger}} = w_{\max} c_t, \qquad R_t^{\text{ledger}} = (1 - w_{\max})c_t. \tag{42}$$

The ledger terms are bookkeeping quantities and must not be confused with Π_{action} and Π_{state}, which are conservative learning penalties and are intentionally not required to sum to c_t. The implementation then (i) logs $(w_{t,\tau}, w_{\max})$, (ii) updates the aligned replay records with soft weights, (iii) trains the delay and ICN heads only on valid labels or aligned hindsight targets, and (iv) reports overlap, TAE, CSR, cost, and reward separately. Oracle lags and SCM labels enter only when the corresponding validation condition is explicitly enabled.

H Failure Analysis

The main failure modes are invalid or missing delay labels; poorly calibrated delay predictions; ICN disagreement with SCM labels for rare actions; multiple actions jointly contributing to one consequence; and distribution shift beyond the states used to calibrate the uncertainty/OOD gate. These failures motivate the diagnostics and conservative scope of our claims.

H.1 MULTI-ACTION AND SEQUENTIAL ATTRIBUTION

The action-centered contrast in Eq. (9) is defined for a single action at a single step. A preliminary diagnostic evaluating a sequential, per-action attribution over a short multi-step horizon (attributing a consequence emitted several steps later back across the intervening actions, rather than to one source action identified by `actual_tau`) recorded substantially weaker agreement with SCM ground truth than the single-step E3 diagnostic reports in Section 7: sign agreement on that preliminary multi-step check was low enough to be consistent with near-chance performance. This measurement predates the current label handling, so it is not yet known whether the weak multi-step agreement reflects a genuine limitation of extending the ICN's single-action contrast to a multi-action, path-specific setting, or whether it is an artifact of the same class of label-generation bug. We report this as an open diagnostic result rather than either a confirmed limitation or a settled conclusion, and we do not fold it into the E3 single-step numbers reported in the main text, since the two measure different estimands. Resolving this is a precondition for any claim that CCPL's causal attribution extends beyond the single-source-action case analyzed in the main text.

I TRAINING ORDER AND SAFEROUTE CASE STUDY

Each episode resets the policy and causal histories, observes transitions, and updates the aligned replay record when delayed feedback identifies a previous source action. Once a batch is available, the update order is: encode stored histories; update the ICN on valid SCM/aligned labels; in the controlled causal-validation configuration, periodically recalibrate against SCM labels; update the delay model; compute y_t^r and y_t^c; update the two critics; update the multiplier; update replay priorities; and softly update target networks. This order keeps history-dependent labels attached to the transition that generated them.

SafeRoute is a separate simulator grid-navigation environment. Its 12-element observation contains normalized agent and goal coordinates, remaining horizon, current and adjacent hazard indicators, normalized Manhattan distance, and the most recently queued cost. The five actions are up, down, left, right, and wait. For candidate cell q_t, the immediate cost is $\tilde{c}_t = \mathbf{1}\{q_t \text{ is hazardous}\} + 0.15\mathbf{1}\{q_t \text{ is outside the grid}\}$. For delay $d > 0$, the environment emits the oldest queued cost and appends $\tilde{c}_t$; pending costs are flushed at termination. The route reward is progress-shaped with a terminal goal bonus and the configured budget is 3. We include the SafeRoute examples to illustrate the delayed queue and action-selection data flow; benchmark results provide the quantitative comparisons.

J OPTIONAL ETHICS STATEMENT

All reported controlled experiments use simulator environments and do not involve human subjects, personal data, or live infrastructure. The domain demonstrations described here are simulations and should not be connected directly to operational systems. A policy satisfying a simulated budget is not certified for physical deployment. Any real deployment would require domain-specific hazard analysis, independent validation, monitoring, fail-safe controls, and appropriate regulatory review.

AI USE STATEMENT

We used an AI language model to assist with manuscript editing and a consistency review of wording and reported claims. We reviewed and verified the final text and are responsible for the technical content, methodology, proofs, experiments, analyses, and claims.