TIME AND CAUSE: RELIABILITY-ADAPTIVE HYBRID CREDIT ASSIGNMENT IN REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Assigning credit to decisions with delayed consequences remains a central challenge in reinforcement learning. Causal credit assignment can reduce policy-gradient variance, but its benefits depend on accurately learned underlying causal mechanisms. In this paper, we study two challenges of this approach: unreliable learned contributions can introduce persistent policy-gradient bias, while ignoring the delay between actions and outcomes can make the relevant causal relationships more difficult to learn. We introduce **H**ybrid **D**elay-aware **C**redit **A**ssignment (HDCA), which addresses both challenges by (i) adaptively interpolating between unbiased temporal and learned causal policy-gradient estimators based on uncertainty in the estimated causal mechanisms and (ii) conditioning credit on the time gap between an action and its outcome. We support these design choices with theoretical analyses that characterize their benefits. Across a diverse suite of existing and newly designed credit-assignment benchmarks, HDCA achieves the best or tied-best final performance while improving sample efficiency and training stability. Mechanistic evaluations of gradient quality and credit-assignment accuracy provide evidence for the sources of these improvements. Finally, targeted experiments empirically validate the theoretical analyses underlying HDCA.

1 INTRODUCTION

Credit assignment asks how much earlier actions contributed to an observed outcome (Minsky, 1961; Pignatelli et al., 2023), a central problem in reinforcement learning (RL) (Sutton et al., 1998). Temporal-difference learning propagates reward information through bootstrapped value estimates, while eligibility traces extend updates to earlier states and actions (Sutton, 1988). Policy-gradient methods use sampled returns to reinforce actions and value baselines to reduce variance (Williams, 1992); Generalized Advantage Estimation balances bias and variance by exponentially weighting TD residuals (Schulman et al., 2015). These methods improve learning signals but retain a temporal heuristic for credit assignment, rather than explicitly measuring each action's contribution to a future outcome. When a delayed reward depends on only a few decisions, irrelevant actions can still receive credit, reducing sample efficiency (Pignatelli et al., 2023). This motivates causal credit assignment methods that estimate each action's influence on future rewarding outcomes.

One class of causal credit assignment approaches is hindsight-based methods (Harutyunyan et al., 2019; Mesnard et al., 2020), which replace this temporal heuristic with a learned *contribution coefficient* measuring how much an action changes the probability of a future rewarding outcome (formally defined in Section 2). The closest work to ours, COCOA (Meulemans et al., 2023), uses these coefficients to weight future rewards in its policy-gradient estimator, assigning credit according to an action's causal influence on each outcome (for a detailed review of prior work, see Appendix A). When the true contribution coefficients are known, the COCOA estimator is unbiased and can reduce variance. In practice, however, the contribution coefficient must be learned from finite, noisy, on-policy data, giving rise to two problems.

(P1) Unmodeled unreliability. The contribution coefficient is estimated using the ratio between the *hindsight* probability of having taken an action in a state, conditioned on observing a future rewarding outcome, and the probability of the corresponding action under the current policy. A learned hindsight model can be inaccurate, particularly for outcomes that are rarely observed. Trusting an inaccurate hindsight model can bias the estimated contribution coefficient and, consequently, the

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.

policy gradient estimate, steering the policy away from valuable but rarely sampled outcomes. Because the data are collected on-policy, this shift can further reduce opportunities to observe those outcomes and correct the estimates, creating a self-reinforcing error. The reliability of learned contributions therefore matters for deciding how much to trust the causal credit signal.

(P2) Pooling over delay. COCOA pools occurrences of the same outcome across action–outcome delays into a single contribution coefficient. If an outcome depends on an earlier action at one delay but occurs independently of it at others, this pooling mixes informative and uninformative events. A richer outcome representation resolves this ambiguity only if it distinguishes those events. Even when the pooled estimator remains unbiased, pooling can weaken the action-dependent signal, making the relevant contributions harder to learn from finite data.

To address these two problems, we propose **H**ybrid **D**elay-aware **C**redit **A**ssignment (HDCA). For P1, HDCA adaptively mixes temporal and causal gradient estimators, using aleatoric and epistemic uncertainties among an ensemble of hindsight models to obtain an adaptive, fine-grained mixing coefficient. This coefficient places more weight on the temporal estimator when the contribution coefficient is unreliable and increasingly relies on the causal estimator as uncertainty decreases. This design is reminiscent of observations in human and animal credit assignment, in which temporal contiguity acts as a provisional cue until a causal mechanism is known (Lagnado & Sloman, 2006). For P2, HDCA conditions contributions on the action–outcome delay, separating informative contributions from unrelated occurrences of the same outcome at other delays. To compensate for the smaller amount of data available at each delay, it learns how much information to share across neighboring delays through kernel smoothing.

Contributions. Each component of HDCA is accompanied by a corresponding theoretical analysis that motivates it, covering the mixing mechanism, the delay conditioning, and the kernel smoothing (Secs. 3.1–3.3). Empirically, HDCA achieves the best or tied-best final performance across diverse benchmarks designed to stress distinct aspects of credit assignment, while generally learning more sample-efficiently and stably than baselines whose relative performance varies substantially across tasks. We probe the source of this sample efficiency through the quality of the estimated policy gradient and the accuracy of the estimated contribution coefficient.

2 Background and Notation

We consider a finite-horizon MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, \mathcal{R}, \mu_0, T)$, with state space $\mathcal{S}$, finite action set $\mathcal{A}$, transition kernel P, reward model $\mathcal{R}$, initial-state distribution μ_0, and horizon $T \geq 2$. A policy π_θ, parameterized by $\theta \in \mathbb{R}^d$, induces trajectories $\tau = (S_t, A_t, R_t)_{t=0}^{T-1}$. We use $\mathcal{T}(s, \pi)$ to denote the distributions over τ under the policy $\pi\,(A|S)$. We are particularly interested in long-term credit assignment, where a rewarding outcome may depend on an action taken much earlier. The objective and its gradient are $J(\theta) := \mathbb{E}_{S_0 \sim \mu_0,\, \tau \sim \mathcal{T}(S_0, \pi_\theta)} \left[\sum_{t=0}^{T-1} R_t\right]$ and $g := \nabla_\theta J(\theta) \in \mathbb{R}^d$. We also use $H_t := T - 1 - t$ to denote the longest possible future delay from *source* time t.

One of the classic approaches to solve delayed reward tasks is called REINFORCE (Williams, 1992):

$$G_R^\gamma = \nabla_\theta^{\text{REINFORCE}} J(\theta) = \sum_{t \geq 0} R_t \sum_{k \leq t} \gamma^{t-k} \nabla_\theta \log \pi(A_k \mid S_k), \quad \gamma \in [0, 1].$$

Although $\gamma < 1$ can reduce variance (Nota & Thomas, 2019), it introduces bias into the gradient estimate. Since we use REINFORCE as an unbiased component of HDCA's gradient estimator, we set $\gamma = 1$ and write $G_R := G_R^1$,. Variance reduction is instead provided by the causal gradient estimator, explained below. We refer to G_R as the *temporal estimator*, reflecting its reliance on temporal ordering for credit assignment.

Hindsight-based credit assignment methods (Harutyunyan et al., 2019; Pignatelli et al., 2023) are built on the concept of the contribution coefficient w. The contribution coefficient w measures how taking action a in state s changes the likelihood of observing outcome u, relative to sampling an action from the current policy, with that policy followed afterward in both cases. In Figure 1, turning right at s_i increases this likelihood ($w > 0$), whereas turning left decreases it ($w < 0$). At s_j, continuing toward u has a positive contribution, while taking the alternative path has a negative

contribution. Thus, actions along the same successful trajectory can receive different credit. While REINFORCE gives positive credit to all actions in a rewarded trajectory, including the red parts that delayed obtaining the reward, causal credit assignment gives negative credit to those parts.

To define the contribution coefficient w, let $U_k \sim U(\cdot \mid S_k, A_k, R_k)$ be a stochastic outcome representation that is predictive of the reward R_k (Meulemans et al., 2023). For a delay $z \in \{1, \dots, H_t\}$, let $p^{\pi}_{t,z}(u \mid s, a) := \mathbb{P}(U_{t+z} = u \mid S_t = s, A_t = a)$ denote the probability of observing the *rewarding outcome* u exactly z steps after taking action a in state s at time t and following the current policy π thereafter. Its policy average is $p^{\pi}_{t,z}(u \mid s) := \sum_b \pi_\theta(b \mid s)\, p^{\pi}_{t,z}(u \mid s, b)$. The *pooled-delay* contribution coefficient is defined as

$$w^{\star,\pi}_t(s,a,u) := \frac{\sum_{z=1}^{H_t} p^{\pi}_{t,z}(u \mid s,a)}{\sum_{z=1}^{H_t} p^{\pi}_{t,z}(u \mid s)} - 1 = \frac{h^{\star,\pi}_t(a \mid s,u)}{\pi_\theta(a \mid s)} - 1. \tag{1}$$

where $h^{\star,\pi}_t(a \mid s, u) := \mathbb{P}(A_t = a \mid S_t = s, U_{K_t} = u)$ is the ground-truth *pooled-delay* hindsight posterior: the conditional probability of having taken action a in state s, given that outcome u is observed at a uniformly sampled future time $K_t \in \{t+1, \dots, T-1\}$, under policy π. The equality of Equation 1 is shown in Lemma E.2.1. For notational simplicity, we suppress the dependence of w, h, $p(u \mid s, a)$, and $p(u \mid s)$ on the policy π throughout. We call w_t and h_t as pooled-delay (or simply *pooled*) since they omit the notion of delay z by summing over all possible delays.

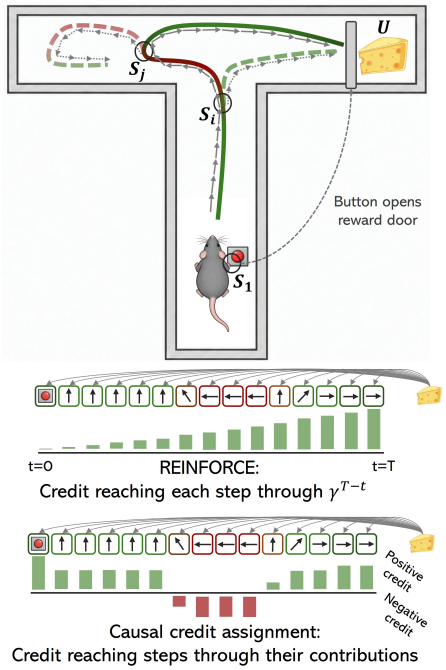


Figure 1: REINFORCE assigns positive credit based on temporal proximity, so essential actions may receive little credit and harmful actions may receive positive credit. Causal credit assignment gives negative credit to harmful actions and positive credit to essential actions through the learned contribution coefficient w. Dashed lines indicate possible continuations of the trajectory for alternative actions in states s_i and s_j; green and red denote beneficial and harmful actions.

The oracle causal pooled gradient estimator (Meulemans et al., 2023) is defined as

$$G^{\star}_C := \sum_{t=0}^{T-1} \sum_{a \in \mathcal{A}} \nabla_\theta \pi_\theta(a \mid S_t) \Big[\frac{\mathbf{1}\{A_t = a\} R_t}{\pi_\theta(a \mid S_t)} + \sum_{z=1}^{H_t} w^{\star}_t(S_t, a, U_{t+z}) R_{t+z} \Big]; \tag{2}$$

replacing $w^{\star}_t$ by a learned $\widehat{w}_t := \widehat{h}_t / \pi_\theta - 1$ gives the learned pooled estimator $\widehat{G}_C$. The oracle and learned pooled estimators share every ingredient except the posterior. Intuitively, the estimator in Equation 2 weights future rewards by each action's contribution to the observed outcome rather than its temporal proximity; it also assigns credit to all possible actions, including those not taken.

COCOA (Meulemans et al., 2023) generalizes hindsight-based causal credit assignment by allowing a general reward-predictive outcome representation, U_k, rather than restricting hindsight conditioning to future states or returns (Harutyunyan et al., 2019). A natural choice for the stochastic mapping U is $U = R$, which is automatically reward-predictive and satisfies the reward-sufficiency property (Appendix E.3.1), yielding the COCOA-reward variant of COCOA. Following COCOA-reward, we use the same representation of the rewarding outcome, setting $U_k = R_k$. We refer to the component of HDCA's gradient estimator that assigns credit using learned action–outcome contribution coefficients as the *causal estimator*.

Lastly, one can train an ensemble of hindsight models rather than a single model. Specifically, an ensemble of size M consists of M hindsight models $\{h_{\phi_m}\}_{m=1}^{M}$, each parameterized by ϕ_m and initialized independently. The ensemble mean is denoted as $\bar{h}_\phi(x) := \frac{1}{M} \sum_{m=1}^{M} h_{\phi_m}(x)$, while the

ensemble variance is defined as $\mathcal{U}^2(x) := \frac{1}{M}\sum_{m=1}^{M}\left\|h_{\phi_m}(x) - \bar{h}_\phi(x)\right\|_2^2$. The variance $\mathcal{U}^2(x)$ can serve as a proxy for epistemic uncertainty at input x, whereas $1 - \|\bar{h}_\phi(x)\|_2^2$, which measures the dispersion of the ensemble-mean posterior, can serve as a measure of aleatoric uncertainty.

3 METHOD

HDCA is organized around the fact that the contribution coefficient w is learned from finite data. In Section 3.1, we define HDCA's delay-conditioned causal estimator and describe how HDCA adapts its reliance on the estimated contribution coefficient (P1); in Section 3.2, we analyze why retaining the delay z between source state S_t and outcome U_{t+z} makes temporally localized coefficients easier to detect[1] (P2); and in Section 3.3, we determine how much information to share across delays.

3.1 RELIABILITY-ADAPTIVE HYBRID CREDIT ASSIGNMENT

The temporal and causal estimators have complementary finite-data behavior. At $\gamma = 1$, the temporal estimator is unbiased, but can have high variance. Conversely, a causal estimator can reduce variance when its contribution coefficient is accurate, but a learned coefficient introduces its own bias. To build HDCA, we combine temporal and causal gradient estimators in an adaptive way.

To characterize the effect of an action at time t on the outcome at time $t+z$, we decompose REINFORCE by source time t and delay z to the reward R_{t+z}, introducing the pairwise (t, z) notation:

$$G_R^{\gamma,(t,z)} := \gamma^z \nabla_\theta \log \pi_\theta(A_t \mid S_t)\, R_{t+z}, \qquad G_R^{\gamma} := \Pi + \sum_{(t,z)\in\mathcal{I}_T} G_R^{\gamma,(t,z)}, \qquad \gamma \in (0,1], \quad (3)$$

where $z \in \{1, \dots, H_t\}$, $\mathcal{I}_T := \{(t,z) : 0 \le t \le T-2,\ 1 \le z \le H_t\}$ and $\Pi := \sum_t \nabla_\theta \log \pi_\theta(A_t \mid S_t) R_t$ is the gradient of the immediate-reward terms. We write $G_R^{(t,z)} := G_R^{1,(t,z)}$.

We also decompose the causal gradient estimator in Equation (2) across delays accordingly. Let $Z := K_t - t$ be a uniform delay random variable and define the delay-conditioned hindsight posterior as $h^\star_{t,z}(a \mid s, u) := \mathbb{P}(A_t = a \mid S_t = s, U_{t+Z} = u, Z = z)$. We define the delay-conditioned contribution coefficient as

$$w^\star_{t,z}(s,a,u) := \frac{p_{t,z}(u \mid s,a)}{p_{t,z}(u \mid s)} - 1 = \frac{h^\star_{t,z}(a \mid s,u)}{\pi_\theta(a \mid s)} - 1. \quad (4)$$

For a learned delay-conditioned posterior $\widehat{h}_{t,z}$, set $\widehat{w}_{t,z}(s,a,u) := \widehat{h}_{t,z}(a \mid s,u)/\pi_\theta(a \mid s) - 1$. Now, we can define the oracle pairwise causal estimator (a decomposition similar to Equation (3)):

$$G_{CD}^{\star,(t,z)} := \sum_{a\in\mathcal{A}} \nabla_\theta \pi_\theta(a \mid S_t) w^\star_{t,z}(S_t, a, U_{t+z}) R_{t+z}, \qquad G^\star_{CD} := \Pi + \sum_{(t,z)\in\mathcal{I}_T} G_{CD}^{\star,(t,z)}. \quad (5)$$

Here, G_{CD} refers to the **C**ausal **D**elay-aware gradient estimator. Replacing $w^\star_{t,z}$ with $\widehat{w}_{t,z}$ gives $\widehat{G}_{CD}^{(t,z)}$ and $\widehat{G}_{CD}$. For $U = R$, the oracle estimator $G^\star_{CD}$ preserves the policy-gradient target and is unbiased (Theorem F.1.3). We analyze the finite-sample benefits of conditioning on the delay (moving from G_C to G_{CD}) in Section 3.2. A **H**ybrid gradient estimator $\widehat{G}_H$ could be formed as a local convex combination of temporal and causal estimators for every source–delay pair (t, z):

$$\widehat{G}_H := \Pi + \sum_{(t,z)\in\mathcal{I}_T} \left[\alpha_{t,z}\widehat{G}_{CD}^{(t,z)} + (1-\alpha_{t,z}) G_R^{(t,z)}\right], \qquad \alpha_{t,z} \in [0,1]. \quad (6)$$

The mixing coefficient $\alpha_{t,z}$ determines the relative contributions of the learned causal and temporal terms. The following theorem, with proof in Theorem E.5.4, motivates HDCA's uncertainty-based mixing mechanism in Equation (8).

Theorem 3.1.1 (Optimal per-pair adaptive convex mixing)**.** *Fix a pair (t, z) and define*

$$\varepsilon_{t,z} := G_R^{(t,z)} - G_{CD}^{\star,(t,z)}, \qquad \Delta_{t,z} := \widehat{G}_{CD}^{(t,z)} - G_{CD}^{\star,(t,z)}.$$

Furthermore, let $V_\varepsilon^{(t,z)}$ denote the variance the causal estimator removes, $V_\varepsilon^{(t,z)} := \mathbb{E}\,\|\varepsilon_{t,z}\|^2 > 0$; and let $V_\Delta^{(t,z)}$ denote the mean-squared error contributed by trusting the learned causal estimator,

[1] We refer to the delay z as the source–outcome delay in some parts to emphasize its interpretation.

$V_\Delta^{(t,z)} := \mathbb{E}\,\|\Delta_{t,z}\|^2$. *Then, the optimal mixing coefficient* $\alpha^\star \in [0,1]^{\mathcal{I}_T}$*, which minimizes the MSE between the hybrid estimator* $\widehat{G}_H$ *and true gradient* g*, is given by*

$$\alpha^\star_{t,z} = \underset{\alpha\in[0,1]^{\mathcal{I}_T}}{\arg\min}\, \mathbb{E}\left[\left\|\widehat{G}_H(\alpha) - g\right\|_2^2\right] = \left(1 + \frac{V_\Delta^{(t,z)}}{V_\varepsilon^{(t,z)}}\right)^{-1}. \tag{7}$$

Intuitively, the optimal mixing coefficient $\alpha^\star_{t,z}$ places more weight on the learned causal estimator $\widehat{G}_{CD}^{(t,z)}$ when the estimation error $V_\Delta^{(t,z)}$ is small relative to the variance reduction $V_\varepsilon^{(t,z)}$ offered by the causal estimator, and shifts toward the temporal estimator $G_R^{(t,z)}$ when $\widehat{G}_{CD}^{(t,z)}$ is unreliable.

Although $\alpha^\star_{t,z}$ comes with an interesting interpretation, the oracle quantities $V_\Delta^{(t,z)}$ and $V_\varepsilon^{(t,z)}$ are not available in practice. However, as we will see, using an ensemble of hindsight models provides surrogate quantities for both $V_\Delta^{(t,z)}$ and $V_\varepsilon^{(t,z)}$. As derived in detail in Appendix E.5, reasonable surrogates for $V_\Delta^{(t,z)}$ and $V_\varepsilon^{(t,z)}$ could be defined as $\widehat{V}_\Delta^{(t,z)} \propto \mathcal{U}^2(x_{t,z})$ and $\widehat{V}_\varepsilon^{(t,z)} \propto 1 - \left\|\bar{h}_\phi(x_{t,z})\right\|_2^2$, respectively, where $x_{t,z} := (t, S_t, U_{t+z}, z)$. Therefore, we train an ensemble of M hindsight models h_{ϕ_m} and also use $\bar{h}_\phi(x_{t,z})$ as the learned hindsight model $\widehat{h}_{t,z}(\cdot \mid S_t, U_{t+z})$.

If we further assume that every action's policy-score vector has the same length, meaning each action pushes the policy parameters equally hard, then $\widehat{V}_\Delta^{(t,z)}$ and $\widehat{V}_\varepsilon^{(t,z)}$ share the same unknown scale. This scale cancels in the ratio (Proposition E.5.6), which leaves

$$\widehat{\alpha}_{t,z} := \frac{\widehat{V}_\varepsilon^{(t,z)}}{\widehat{V}_\varepsilon^{(t,z)} + \widehat{V}_\Delta^{(t,z)}}, \qquad \underbrace{\widehat{V}_\varepsilon^{(t,z)} := 1 - \left\|\bar{h}_\phi(x_{t,z})\right\|_2^2}_{\text{aleatoric}}, \qquad \underbrace{\widehat{V}_\Delta^{(t,z)} := \tfrac{M}{M-1}\mathcal{U}^2(x_{t,z})}_{\text{epistemic}}. \tag{8}$$

The practical mixing coefficient $\widehat{\alpha}_{t,z}$ adjusts the causal weight according to the ratio of the epistemic and aleatoric surrogates, assigning greater weight to the temporal estimator when this ratio is larger. Equation (8) has no tuned parameters and is invariant to a global rescaling of the rewards and of the policy scores (Proposition E.5.6); comparison with an annealed mixing coefficient (Appendix B.5) further demonstrates the importance of this tuning-free design. For probability-vector ensemble predictions, $\mathcal{U}^2 \le 1 - \|\bar{h}_\phi\|_2^2$, which implies $\widehat{\alpha}_{t,z} \ge (M-1)/(2M-1)$ (Proposition E.5.6). Thus, the practical rule adjusts the temporal contribution while always retaining a positive weight for the causal estimator. Moreover, each ensemble member is weakly regularized toward its initialization, which helps preserve diversity, and hence informative disagreement, on poorly supported inputs.

We elaborate on the theoretical implications of the hybrid gradient estimator in Equation (6) in Appendices E.4–E.9. In summary, we show that the bias of the hybrid estimator $\widehat{G}_H$ decomposes into two sources: (i) the bias carried by the learned causal estimator $\widehat{G}_{CD}^{(t,z)}$ and (ii) the bias induced by a data-dependent mixing coefficient (for instance, our practical mixing coefficient $\widehat{\alpha}_{t,z}$). We then bound this bias and show that, as $\bar{h}_\phi$ becomes more accurate and $\widehat{\alpha}_{t,z}$ approaches one, both sources of bias vanish. We also establish a result for an arbitrary mixing coefficient α, showing that $\widehat{G}_H$ has lower variance than G_R whenever $\alpha V_\Delta \le (2-\alpha)V_\varepsilon$.

3.2 Delay Conditioning Gains

We now compare the delay-conditioned causal estimator $G^\star_{CD}$ in Equation (5) with the pooled-delay estimator $G^\star_C$ in Equation (2). The oracle estimator $G^\star_{CD}$ is unbiased (Theorem F.1.3); its variance depends on the structure of the task, and neither the delay-conditioned $G^\star_{CD}$ nor the pooled estimator $G^\star_C$ always has lower variance (Proposition F.3.2). Appendix F.2 shows that delay conditioning yields a more accurate contribution coefficient in the infinite-data regime. However, the contribution coefficient is learned from limited data, so analyzing the finite-data regime is more relevant.

To quantify the effect of pooling, for a source time t, state s, and outcome u, let

$$\rho_{t,z}(s,u) := \frac{p_{t,z}(u \mid s)}{\sum_{j=1}^{H_t} p_{t,j}(u \mid s)} = p(z \mid x), \qquad x = (t,s,u), \tag{9}$$

denote the share of the outcome's occurrence probability carried by delay z. It is also the expected fraction of hindsight tuples with input x that come from delay z. The pooled coefficient is the ρ-weighted average of the delay-conditioned ones, $w_t^\star(s,a,u) = \sum_z \rho_{t,z}(s,u)\, w_{t,z}^\star(s,a,u)$.

Consider the setting in which the causal influence of an action is localized at a single delay $z^\star$, where the advantage of delay conditioning over delay pooling is clearest. This advantage becomes smaller as the causal influence spreads across delays. In this localized setting, we have $w_t^\star(s,a,u) = \rho_{t,z^\star}(s,u)\, w_{t,z^\star}^\star(s,a,u)$. In other words, pooling mixes the occurrences at $z^\star$ with unrelated occurrences of the same outcome at other delays, which dilutes the action's apparent contribution. The more occurrences fall outside $z^\star$, the stronger the dilution.

When the coefficient must be learned from finite data, this weaker signal is harder to distinguish from estimation noise. We refer to this as the problem of *detectability*: whether an estimated contribution coefficient can be reliably distinguished from zero, the value corresponding to no contribution. We measure detectability with the detectability signal-to-noise ratio of an estimator $\widehat{w}$, $\mathrm{SNR}_{\mathrm{D}}(\widehat{w}) := \mathbb{E}[\widehat{w}]^2 / \operatorname{Var}(\widehat{w})$. The following theorem compares this quantity for the pooled and delay-conditioned estimators.

Theorem 3.2.1 (Delay-conditioned detectability under finite data)**.** *Fix (t,s,a,u) with $w_{t,z^\star}^\star(s,a,u) \neq 0$, $w_{t,z}^\star(s,a,u) = 0$ for all $z \neq z^\star$, and $\pi_\theta(a \mid s)$, $h_{t,z^\star}^\star(a \mid s,u) \in (0,1)$. Consider n hindsight tuples with input $x = (t,s,u)$ and conditionally independent actions. Their delays $Z_{1:n}$ are given, and exactly $n\,\rho_{t,z^\star}(s,u) \geq 1$ of them are equal to $z^\star$. Let $\widehat{w}_t(s,a,u)$ and $\widehat{w}_{t,z^\star}(s,a,u)$ be the pooled and delay-conditioned empirical contribution coefficients computed from these tuples. Then, with the moments in SNR$_D$ taken conditionally on $Z_{1:n}$,*

$$\frac{\mathrm{SNR}_D\big(\widehat{w}_{t,z^\star}(s,a,u)\big)}{\mathrm{SNR}_D\big(\widehat{w}_t(s,a,u)\big)} = 1 + \frac{\big(1-\rho_{t,z^\star}(s,u)\big)\,\pi_\theta(a \mid s)\big(1-\pi_\theta(a \mid s)\big)}{\rho_{t,z^\star}(s,u)\,h_{t,z^\star}^\star(a \mid s,u)\big(1-h_{t,z^\star}^\star(a \mid s,u)\big)}, \tag{10}$$

which exceeds one if and only if $\rho_{t,z^\star}(s,u) < 1$, and is $\Theta\big(\rho_{t,z^\star}(s,u)^{-1}\big)$ as $\rho_{t,z^\star}(s,u) \to 0$. The proof is in Theorem F.4.1.

The higher $\mathrm{SNR}_{\mathrm{D}}$ of $\widehat{w}_{t,z^\star}$ compared to $\widehat{w}_t$ has a simple source. Write $\rho := \rho_{t,z^\star}(s,u)$ and recall that $\mathrm{SNR}_{\mathrm{D}}$ is a squared mean divided by a variance. Pooling uses all n tuples but dilutes the mean to $\rho\, w_{t,z^\star}^\star(s,a,u)$, so its $\mathrm{SNR}_{\mathrm{D}}$ shrinks by ρ^2. Delay conditioning keeps the full mean but averages only the $n\rho$ tuples at delay $z^\star$, so its variance grows by $1/\rho$ and its $\mathrm{SNR}_{\mathrm{D}}$ shrinks only by ρ (Theorem F.4.1). Dilution also makes contributions easier to miss. Because pooling shrinks the contribution, even a small estimation error can hide it. Delay conditioning keeps the contribution at the original scale, so tolerates larger errors (Proposition F.4.3). Since posterior error transfers linearly to causal-estimator bias (Theorem E.7.1), this affects credit assignment and sample efficiency.

3.3 Kernel Smoothing Across Neighboring Delays

As we saw, separating delays improves detectability but reduces the number of hindsight tuples available at each delay, so a fixed global choice between delay conditioning and pooling is unnecessarily rigid. To manage this trade-off, we share information across nearby delays as follows: For each query (x,z) with $x = (t,s,u)$, HDCA uses a bandwidth network with parameters ψ_m per hindsight model h_{ϕ_m} to predict a bandwidth $\sigma_m := \sigma_{\psi_m}(x,z) > 0$. An observation from delay z' then contributes to the loss for delay z with the normalized weight $\omega_{\sigma_m}(z' \mid x,z) := p(z' \mid x)\, K_{\sigma_m}(z,z') \big/ \sum_{z''} p(z'' \mid x)\, K_{\sigma_m}(z,z'')$, where K_{σ_m} is a nonnegative kernel and $p(z' \mid x)$ is defined in Equation (9). HDCA then trains the parameters ϕ_m and ψ_m of each ensemble member by minimizing

$$\mathcal{L}_m(\phi_m,\psi_m) := \mathbb{E}_{(X,Z)}\Big[\sum_{z'} \omega_{\sigma_m}(z' \mid X,Z)\, \mathbb{E}_{A\sim h_{t,z'}^\star(\cdot\mid S_t,\,U_{t+Z})}\big[-\log h_{\phi_m}(A \mid X,Z)\big]\Big]. \tag{11}$$

The inner expectation averages over actions taken at (t,S_t) on trajectories that observe the same outcome z' steps later. The kernel loss is therefore a weighted average of delay-conditioned losses. The hindsight and bandwidth networks, h_{ϕ_m} and σ_{ψ_m} respectively, are trained in alternation to prevent degenerate solutions that would otherwise minimize the loss. Pseudocode is in Appendix J.

We further discuss the theoretical implications of kernel smoothing in Appendix G. In this appendix, we show that the objective in Equation (11) targets a weighted average of the posteriors at neighboring delays, $\sum_{z'} \omega_{\sigma_m}(z' \mid x,z)\, h_{t,z'}^\star(\cdot \mid s,u)$. The resulting contribution coefficient approaches the

delay-conditioned coefficient $w^\star_{t,z}$ as $\sigma \downarrow 0$ and the pooled coefficient $w^\star_t$ as $\sigma \to \infty$. HDCA can thus interpolate locally between these two extremes (Appendix G.2). Sharing observations across delays gives each posterior estimate more hindsight tuples, reducing variance (Proposition G.3.3), but biases $\widehat{w}_{t,z}$ when neighboring delays have different contribution coefficients. Smoothing thus helps when the variance it removes outweighs the squared bias it adds (Theorem G.4.1); the learned bandwidth lets HDCA adjust this trade-off locally (Appendix B.4).

4 EXPERIMENTS

We first introduce the baselines and benchmarks, then use two diagnostic tasks to illustrate how P1 and P2 harm COCOA but not HDCA. Our analysis evaluates the learning signal at two levels. For the gradient, we use the policy-gradient signal-to-noise ratio, $\mathrm{SNR} = \|g\|_2^2 \big/ \mathbb{E}_{\tau\sim\pi_\theta}\big[\|\widehat{g}(\tau) - g\|_2^2\big]$, where $\widehat{g}(\tau)$ is an estimator's gradient from one on-policy trajectory; a higher value means fewer trajectories are needed for an accurate gradient (Meulemans et al., 2023). For the contribution coefficient, we use the Hindsight Credit Reconstruction error, $\mathrm{HCR} = \sum_{s\in\mathcal{S}_{\mathrm{crit}},a,z} \mu^\pi(s,a)\big(\widehat{C}_z(s,a) - C^\star_z(s,a)\big)^2 \big/ \sum_{s\in\mathcal{S}_{\mathrm{crit}},a,z} \mu^\pi(s,a)\, C^\star_z(s,a)^2$. It compares the credit $\widehat{C}_z(s,a)$ that a method assigns to action a at a critical state s for the reward z steps later with its ground truth $C^\star_z(s,a)$, weighted by state–action visitation μ^π; a lower value means a more accurate delay-resolved contribution coefficient. Full definitions of the metrics, including gradient bias and variance, are in Appendix I.4, with ground-truth quantities computed as in Meulemans et al. (2023) (Appendix H). Section 4.3 tests whether HDCA's adaptive mixing of temporal and causal estimators improves learning across tasks with different credit-assignment structures, and finally Section 4.4 extends results to a more challenging task.

In the appendix, we extend our analysis to provide more insight into the improved performance. In Appendix B.2, we validate the improved SNR and HCR across a broad range of tasks. In Appendix B.1, we analyze the dynamics of the mixing coefficient and estimated uncertainties, and their effects on bias, variance, and SNR. In Appendix B.5, we compare our adaptive mixing strategy with optimal mixing and an annealing-based strategy, and observe close alignment between our mixing coefficient and the optimal one. In Appendix B.3, we show that delay conditioning improves SNR relative to the pooled-delay variant. Finally, in Appendix C, we provide an extensive ablation study of different variants of HDCA and COCOA, as well as the effect of ensemble size.

4.1 TASKS AND BASELINES

We include the original conveyor task of Meulemans et al. (2023), which we call `Short/Long Key-door`: the agent must pick up a key to collect a treasure at the end of a conveyor with stochastic intermediate rewards. Because most methods saturate on it, we also design five new conveyor benchmarks that systematically test credit assignment across different regimes. As in Meulemans et al. (2023), all environments have small discrete action spaces for interacting with objects (e.g., picking up a key), automatic transitions between cells, full observability, and easy exploration, which isolates credit assignment from exploration and representation learning.

`Ordered-keys` requires collecting two keys in a specific order. `Colored-keys` requires selecting the key matching an explicit context color, whereas `Multicolored-keys` makes this context implicit through a fixed ranking of colors. `XOR-keys` rewards collecting exactly one of two keys. In `Poisoned-apples`, the first poisoned apple yields an immediate reward but silently forfeits the final treasure. Full specifications, motivating psychology and cognitive-science experiments, and the credit-assignment challenges targeted by each task appear in Appendix I. A more complex task, adapted from Montezuma's Revenge, is described in Appendix I.3. This variant preserves the credit-assignment challenge while reducing representation-learning and exploration demands.

We compare against undiscounted REINFORCE (Williams, 1992) and its variant with a learned value baseline (Advantage), trajectory-wise control variates (TrajCV) (Cheng et al., 2020), HCA+ (Harutyunyan et al., 2019), a state–action value critic (Q-critic), and both variants of COCOA (Meulemans et al., 2023), named COCOA-reward and COCOA-feature. In COCOA-feature, U is obtained from a network that predicts a reward given a state and an action. All methods share the same policy architecture and data budget, so differences reflect the credit-assignment mechanism rather than policy capacity (for a comparison of the baselines, refer to Appendix I.1). We report the

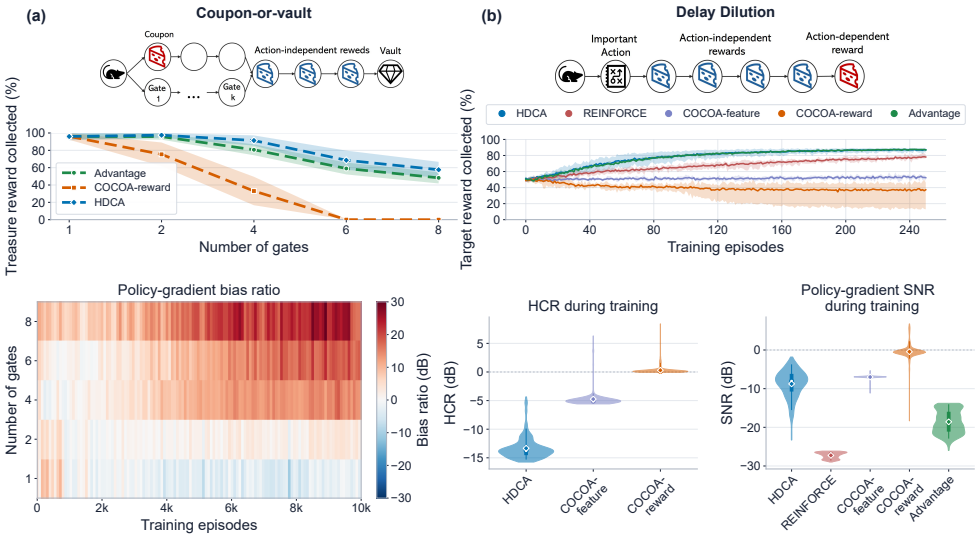


Figure 2: **Diagnostic tasks for the two credit-assignment challenges.** (a) `Coupon-or-vault` probes unreliable contribution estimates as valuable rewards become harder to discover; positive bias ratios indicate greater bias in COCOA-reward than in HDCA. (b) `Delay Dilution` probes pooling across informative and uninformative delays, reporting reward collection, HCR, and policy-gradient SNR.

mean and 95% bootstrap intervals over 5 seeds. A list of hyperparameters, the tuning procedure, and a comprehensive analysis of hyperparameter sensitivity are provided in Appendix I.5.

4.2 Diagnostic Tasks

Our first experiments are diagnostic tasks designed to isolate the two challenges from Section 1: unreliable contribution estimates (P1) and pooling across delays (P2). In `Coupon-or-vault` (P1), the agent chooses between a small immediate reward and a larger delayed reward that requires the correct action at each of several binary gates; more gates make the larger reward harder to discover. As shown in Figure 2a, COCOA-reward's performance collapses as the number of gates grows, whereas HDCA's degrades much more slowly. The heatmap shows the ratio of COCOA-reward's policy-gradient bias to HDCA's (in dB): the ratio is large and persistent at higher gate counts, reflecting how hard it is to learn reliable contributions when valuable outcomes are rarely observed. In `Delay Dilution` (P2), an earlier action influences a reward at one delay, while identical rewards occur independently of that action at other delays. Pooling these occurrences dilutes the informative contribution coefficient. As shown in Figure 2b, HDCA collects the target reward more often than COCOA-reward and keeps improving throughout training, as do the temporal methods Advantage and REINFORCE. The violin plots may explain why: HDCA has the lowest HCR, i.e., an accurate contribution coefficient, while keeping a strong learning signal for the policy (high SNR).

4.3 HDCA Improves Sample Efficiency, Final Performance, and Training Stability

Figure 3 compares performance of methods across conveyor-based tasks. HDCA attains the best or tied-best final performance on all six tasks, and reaches it from fewer episodes. Also, it does not collapse or oscillate, unlike the baselines. Complete results, including performance on the contextual-bandit task `Stochastic-interleaving` (Meulemans et al., 2023) and the two diagnostic tasks `Coupon-or-vault` and `Delay Dilution`, are reported in Appendix B.6.

We then investigate how the baselines fail: Causal estimators are strongest where credit is sparse, long-range and structurally identifiable. COCOA-feature is the only baseline to solve `Ordered-keys` and performs well by the end of the experiment on `Colored-keys`, tasks on which the temporal estimators (Advantage and REINFORCE) perform poorly. The ordering reverses when the contribution coefficient is hard to learn on-policy. On `XOR-keys`, where credit is non-additive in the action history, COCOA-feature climbs to $\approx 40\%$ and then declines, while Advantage, whose recency prior matches the task, is near-optimal. Each baseline thus inherits both the

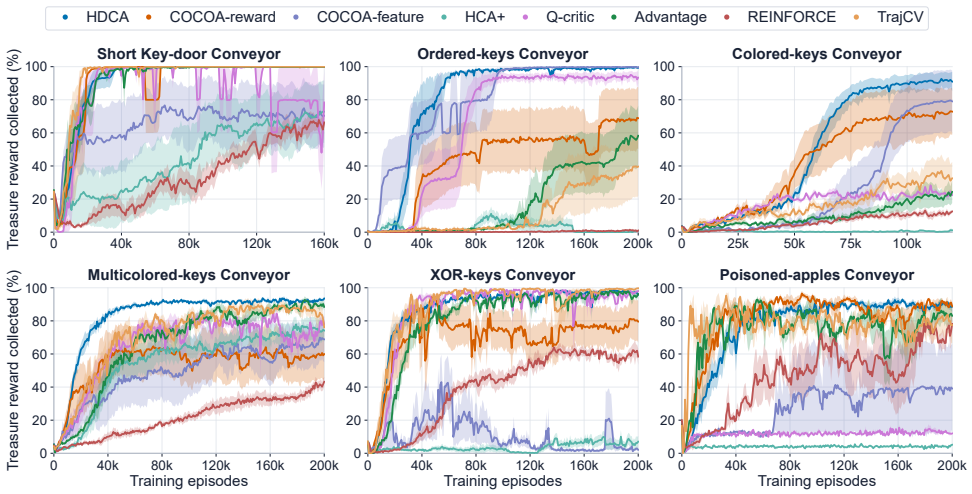


Figure 3: **Performance across conveyor-based tasks.** HDCA is the only method that achieves the best or tied-best final performance across all environments, while the performance of the baselines varies across tasks. HDCA generally learns faster and more stably. Shaded regions denote 95% bootstrap intervals over five seeds.

strengths and the failure modes of its fixed principle, and which principle is right is a property of the task, not something known in advance. Using the local uncertainty surrogates in Equation (8), HDCA adjusts the relative contributions of causal and temporal estimators, increasing the temporal weight when ensemble disagreement is larger relative to the aleatoric surrogate.

4.4 A More Complex Task

As in our other experiments, we focus on credit assignment and separate it from exploration and representation learning. Following this, we extend our evaluation to the more complex Montezuma's Revenge. To remove representation learning, we use an abstract symbolic version of the game with an 84-dimensional state space. To ease exploration, we initialize all policies by behavior cloning on a small set of expert demonstrations and then fine-tune them with each RL algorithm, so that every method sees a few successful trajectories from the start. Value functions and hindsight models are also warm-started before RL fine-tuning, so they do not start from scratch while the policy already benefits from behavior cloning. Full details and hyperparameters are in Appendix I.

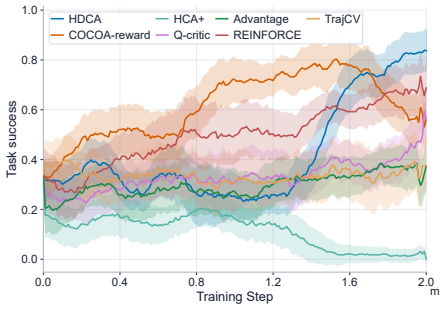


Figure 4: **Performance on Montezuma's Revenge.** HDCA achieves the highest final task success, after a slower initial improvement, while COCOA-reward reaches strong intermediate performance but declines later.

Figure 4 shows that HDCA attains the strongest final performance on the Montezuma's Revenge task. COCOA-reward improves more quickly early in training and reaches high success, but its performance declines later. HDCA instead improves more gradually, then rises sharply in the later stages and ultimately surpasses baselines. REINFORCE also improves steadily but remains below HDCA, while the other methods plateau at lower success levels or deteriorate. This suggests that HDCA's adaptive mixing mechanism remains effective in the more complex setting considered here.

5 Conclusion

Our results suggest that temporal and causal credit assignment are complementary rather than competing. HDCA treats the learned contribution coefficient as an estimate whose reliability must itself be assessed, rather than as ground truth. Based on this, HDCA adaptively weights temporal and learned causal credit according to estimated hindsight reliability, and resolves credit by delay so that localized contributions remain detectable. Extending HDCA to richer outcome encodings and high-dimensional observations remains important future work.

AI USE STATEMENT

We follow the ICLR 2027 AI Policy for Authors, which divides AI-assisted tasks into two categories: tasks that must be disclosed if AI was used (required disclosure), and tasks whose disclosure is encouraged (recommended disclosure). We report our use of generative AI tools under each category below.

Tasks with required disclosure. In this work, we used generative AI tools to implement methods and to assist with proofs. (i) Our code builds on the public implementation of COCOA (Meulemans et al., 2023); we used AI tools to implement parts of HDCA and of the benchmark tasks (Section 4 and Appendix I). The authors reviewed and tested all AI-generated code before using it in any experiment. (ii) The authors developed all theorem statements, assumptions, and proofs in Appendices D–G. We then used AI tools to check the proofs for errors and for consistent notation. The authors verified every suggested correction before incorporating it. We did not use generative AI tools to develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proofs, propose or refine hypotheses, design or give feedback on the research methodology or experiments, or interpret results. Generating, cleaning, or reformatting datasets, qualitative data analysis, and translation are not applicable to this work, since all experiments use online interaction with simulated environments.

Tasks with recommended disclosure. Additionally, we used generative AI tools to (i) identify relevant literature, both on credit assignment in reinforcement learning (Appendix A) and on the psychology and cognitive-science paradigms that motivate our tasks (Appendix I.2); the authors read every identified work in full before citing it; (ii) edit text written by the authors to improve concision, clarity, and grammar; and (iii) review successive drafts of the paper and suggest improvements to its structure, with the authors deciding which suggestions to adopt.

We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have taken the following steps to make our results reproducible. **Code.** The supplementary material contains our full source code, anonymized, in two versions. The first implements HDCA, all baselines, and all tasks except Montezuma's Revenge. The second adds the Montezuma's Revenge benchmark (Section 4.4), together with its demonstration-collection and behavior-cloning pretraining pipeline. Both versions build on the public COCOA implementation (Meulemans et al., 2023). They include the configuration files and launch scripts for our experiments, with the random seeds used, and unit tests for the environments, estimators, and evaluation metrics. We will release the code publicly upon publication. **Method and theory.** Section 3 describes HDCA, and Appendix J gives its pseudocode. Appendix D states all definitions and assumptions, and Appendices E–G contain complete proofs of all theoretical results. **Experiments.** Appendix I.1 describes the baselines, and Appendix I.2 specifies every task. For Montezuma's Revenge, Appendix I.3 describes the state representation, the demonstrations, and the pretraining protocol. The demonstrations are generated by a scripted controller included in the code, so no external data is required. Appendix H explains how the ground-truth quantities used in our mechanistic analyses are computed exactly by dynamic programming, and Appendix I.4 defines the evaluation metrics. Appendix I.5 reports all hyperparameters, the tuning procedure, and a sensitivity analysis. Results are reported as means with 95% bootstrap intervals over five seeds (Section 4).

REFERENCES

Coreen Adamson, T Mary Foster, and James SA McEwan. Delayed matching to sample: the effects of sample-set size on human performance. *Behavioural processes*, 49(3):149–161, 2000.

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

Peter E Bryant and Thomas Trabasso. Transitive inferences and memory in young children. *Nature*, 1971.

Jacob Buckman, Danijar Hafner, George Tucker, Eugene Brevdo, and Honglak Lee. Sample-efficient reinforcement learning with stochastic ensemble value expansion. *Advances in neural information processing systems*, 31, 2018.

Ching-An Cheng, Xinyan Yan, and Byron Boots. Trajectory-wise control variates for variance reduction in policy gradient methods. In *Conference on Robot Learning*, pp. 1379–1394. PMLR, 2020.

Russell M Church and Marvin Z Deluty. Bisection of temporal intervals. *Journal of Experimental Psychology: Animal Behavior Processes*, 3(3):216, 1977.

Esraa Elelimy, Brett Daley, Andrew Patterson, Marlos C Machado, Adam White, and Martha White. Deep reinforcement learning with gradient eligibility traces. *arXiv preprint arXiv:2507.09087*, 2025.

John Garcia and Robert A Koelling. Relation of cue to consequence in avoidance learning. *Psychonomic science*, 4(1):123–124, 1966.

W James Greville and Marc J Buehner. Temporal predictability facilitates causal learning. *Journal of Experimental Psychology: General*, 139(4):756, 2010.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

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

Nicolas Heess, Gregory Wayne, David Silver, Timothy Lillicrap, Tom Erez, and Yuval Tassa. Learning continuous control policies by stochastic value gradients. *Advances in neural information processing systems*, 28, 2015.

Ralph Hertwig, Greg Barron, Elke U Weber, and Ido Erev. Decisions from experience and the effect of rare events in risky choice. *Psychological science*, 15(8):534–539, 2004.

Chia-Chun Hung, Timothy Lillicrap, Josh Abramson, Yan Wu, Mehdi Mirza, Federico Carnevale, Arun Ahuja, and Greg Wayne. Optimizing agent behavior over long time scales by transporting value. *Nature communications*, 10(1):5223, 2019.

Vijay Konda and John Tsitsiklis. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

Harald Lachnit and Klaus Lober. What is learned in patterning discriminations? further tests of configural accounts of associative learning in human electrodermal conditioning. *Biological Psychology*, 56(1):45–61, 2001.

David A Lagnado and Steven A Sloman. Time as a guide to cause. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 32(3):451, 2006.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Kennon A Lattal. Delayed reinforcement of operant behavior. *Journal of the Experimental Analysis of Behavior*, 93(1):129–139, 2010.

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

Alexander Meulemans, Simon Schug, Seijin Kobayashi, Gregory Wayne, et al. Would i have gotten that reward? long-term credit assignment by counterfactual contribution analysis. *Advances in Neural Information Processing Systems*, 36:68685–68735, 2023.

Marvin Minsky. Steps toward artificial intelligence. *Proceedings of the IRE*, 49(1):8–30, 1961. doi: 10.1109/JRPROC.1961.287775.

Chris Nota and Philip S Thomas. Is the policy gradient a gradient? *arXiv preprint arXiv:1906.07073*, 2019.

Vihang P Patil, Markus Hofmarcher, Marius-Constantin Dinu, Matthias Dorfer, Patrick M Blies, Johannes Brandstetter, Jose A Arjona-Medina, and Sepp Hochreiter. Align-rudder: Learning from few demonstrations by reward redistribution. *arXiv preprint arXiv:2009.14108*, 2020.

Andrew Patterson, Samuel Neumann, Martha White, and Adam White. Empirical design in reinforcement learning. *Journal of Machine Learning Research*, 25(318):1–63, 2024.

Eduardo Pignatelli, Johan Ferret, Matthieu Geist, Thomas Mesnard, Hado van Hasselt, Olivier Pietquin, and Laura Toni. A survey of temporal credit assignment in deep reinforcement learning. *arXiv preprint arXiv:2312.01072*, 2023.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3 (1):9–44, 1988.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

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

CONTENTS

1 Introduction **1**

2 Background and Notation **2**

3 Method **4**

3.1 Reliability-Adaptive Hybrid Credit Assignment 4

3.2 Delay Conditioning Gains 5

3.3 Kernel Smoothing Across Neighboring Delays 6

4 Experiments **7**

4.1 Tasks and Baselines 7

4.2 Diagnostic Tasks 8

4.3 HDCA improves sample efficiency, final performance, and training stability 8

4.4 A More Complex Task 9

5 Conclusion **9**

A Related Work **14**

B Further Results and Analysis **16**

B.1 The adaptive mixing balances bias and variance throughout training 16

B.2 The SNR and HCR are more favorable in HDCA 17

B.3 Delay conditioning leads to higher SNR 17

B.4 Kernel smoothing is used when useful and otherwise reverts to the delay-conditioned regime 18

B.5 HDCA adaptive mixing comparison with optimal and annealing mixing 18

B.6 Extensive Results 20

C Ablation Study **21**

C.1 Ablated Variants 21

C.2 Different Ensemble Sizes 22

D Definitions, Assumptions, Conventions, and Notation **23**

E Hybrid Gradient Estimator **25**

E.1 Discounting bias of the temporal estimator 25

E.2 Pooled hindsight identities 26

E.3 Unbiasedness of the vanilla pooled-delay estimator 27

E.4 Adaptive per-pair mixing: exact global projection-gap analysis 28

E.5 Estimating the optimal mixing: pair-local derivation 30

E.6 Bias of adaptive mixing 35

E.7 Propagation of hindsight-posterior error 35
E.8 Hindsight consistency and vanishing adaptive bias 37
E.9 Bias bounds relative to the temporal and causal estimators 39

F Delay-Conditioned Causal Credit **41**
F.1 Delay conditioning preserves the policy-gradient target 41
F.2 When delay provides strictly richer predictive information 42
F.3 Local oracle mean and variance under pooling and delay conditioning 43
F.4 What delay conditioning does buy: detectability of localized influence 44

G Kernel-Smoothed Delay Conditioning **47**
G.1 Population target of the kernelized loss 47
G.2 The bandwidth interpolates between delay-conditioned and delay-pooled estimators 48
G.3 Smoothing bias and effective sample size 48
G.4 When smoothing helps: variance reduction against squared bias 50

H Ground-Truth Policy Gradient **52**

I Experimental Details and Hyperparameters **52**
I.1 Baselines 52
I.2 Environment Details 53
I.3 Atari: Montezuma's Revenge 56
I.4 Metrics 57
I.5 Hyperparameters 58

J Algorithm Pseudocode **59**

A Related Work

Credit assignment in RL has traditionally focused on how rewards should be propagated back to earlier decisions in a trajectory. Monte Carlo policy-gradient methods such as REINFORCE (Williams, 1992), for example, use the return-to-go to assign future rewards to all actions that came before them. Actor–critic methods follow the same recency principle, while they use a learned value function to produce a lower-variance learning signal (Konda & Tsitsiklis, 1999). To improve credit assignment, temporal-difference methods and eligibility traces, including TD(λ), provide a middle ground by combining short- and long-horizon updates (Sutton, 1988). Generalized Advantage Estimation (GAE) (Schulman et al., 2015) builds on this idea for policy-gradient methods by combining multi-step advantage estimates with exponential weighting, allowing the bias–variance trade-off to be adjusted. A complementary line of work reduces policy-gradient variance without fundamentally changing which earlier decisions receive credit. Trajectory-wise control variates, for example, exploit correlations between gradient contributions along the same trajectory to construct more effective control variates (Cheng et al., 2020). While these methods improve the bias–variance trade-off of the policy gradient, they are all built upon recency bias: a reward is propagated to earlier actions because they precede it, rather than because a particular action is inferred to have contributed to that particular outcome.

Another line of work targets long delays more explicitly by restructuring the temporal path between an influential decision and its eventual reward. RUDDER (Arjona-Medina et al., 2019) learns a

decomposition of the return and redistributes delayed rewards toward state–action pairs. Align-RUDDER (Patil et al., 2020) develops this idea for settings in which only a small number of successful demonstrations are available, using sequence alignment to identify common event structure and redistribute reward accordingly. Temporal Value Transport (TVT) (Hung et al., 2019) takes a different approach: an attention-based memory mechanism identifies past events that are later retrieved as relevant and transports value back to those events, effectively creating shortcuts across long temporal gaps. These methods demonstrate that long-term credit can be improved by identifying salient events instead of relying exclusively on temporal propagation. Their objective, however, differs from ours. RUDDER and Align-RUDDER reorganize the reward signal, while TVT transports value through learned memory associations. HDCA instead retains the original rewards and estimates how a specific earlier action contributes to a specific later rewarding outcome, while explicitly accounting for uncertainty in that learned attribution. The distinction is central to the present work, whose estimator combines a temporally grounded REINFORCE estimator with a mathematically grounded causal contribution signal, rather than redistributing the environment reward itself.

Eligibility traces provide a classical mechanism for accelerating temporal credit assignment by propagating TD errors to multiple preceding states or actions, with the trace-decay parameter λ controlling how rapidly this influence decreases with temporal distance (Sutton, 1988). This perspective is closely related to the multi-step estimators used in modern policy-gradient methods: in particular, GAE can be viewed as an exponentially weighted combination of TD residuals. Recent work has brought eligibility-trace ideas more explicitly into deep RL. Elelimy et al. (2025) extend Gradient TD methods to multistep credit assignment through λ-returns, deriving both forward-view and backward-view formulations. Their Gradient PPO modifies PPO's value-learning component to use these gradient-corrected multistep updates, while their streaming algorithms admit a backward-view implementation using eligibility traces. These developments demonstrate that trace-based temporal credit assignment can be incorporated effectively into modern deep RL algorithms. Nevertheless, like GAE and other exponentially weighted multi-step methods, eligibility traces primarily determine credit according to temporal proximity rather than estimating whether a particular earlier action actually contributed to a particular later outcome; HDCA targets this latter distinction explicitly.

Hindsight-based methods have the closest philosophy to our method. Hindsight Credit Assignment (HCA), including the HCA+ variant, assigns credit to a past decision according to the posterior probability of that decision given the information observed later in the trajectory (Harutyunyan et al., 2019). This replaces purely temporal attribution with a learned association between past actions and subsequent outcomes. Counterfactual Credit Assignment (CCA) (Mesnard et al., 2020) similarly uses hindsight, but constructs future-conditioned value functions that support trajectory-specific counterfactual estimates. To control the bias that can arise from conditioning on the future, CCA constrains the hindsight information so that it does not reveal the agent's action. COCOA (Meulemans et al., 2023) is most directly related to HDCA. Building on HCA, it argues that conditioning on rewarding states can introduce spurious credit and instead measures an action's contribution with respect to reward-predictive outcomes. Its reward-based and learned-feature variants, COCOA-reward and COCOA-feature, can substantially reduce variance when the learned contribution coefficient accurately captures the relevant counterfactual relationship.

The way past experience is selected for learning can also affect how efficiently credit propagates. Prioritized Experience Replay (PER) (Schaul et al., 2015) samples transitions considered more informative, typically those with large TD errors, more frequently than under uniform replay. Although PER does not explicitly estimate which earlier action caused a later reward, it can be interpreted as an indirect mechanism for improving credit assignment. Concentrating updates on transitions with large Bellman errors can accelerate the propagation of newly observed reward information through bootstrapped value estimates. A different family of approaches obtains richer long-horizon learning signals by learning a model of the environment and propagating value or reward information through that model. Stochastic Value Gradients (SVG) (Heess et al., 2015), for example, formalize policy optimization by backpropagating value gradients through differentiable learned dynamics, providing a continuum between model-free and model-based policy-gradient methods. More recent world-model approaches such as Dreamer learn latent dynamics and optimize behavior by propagating gradients of learned state values through imagined trajectories (Hafner et al., 2019); the Dreamer line has subsequently been extended to broad-domain agents such as DreamerV3 (Hafner et al., 2023). Such approaches can transmit learning signals across long horizons, but they require learning and repeatedly using a sufficiently accurate predictive model of environment dynamics and

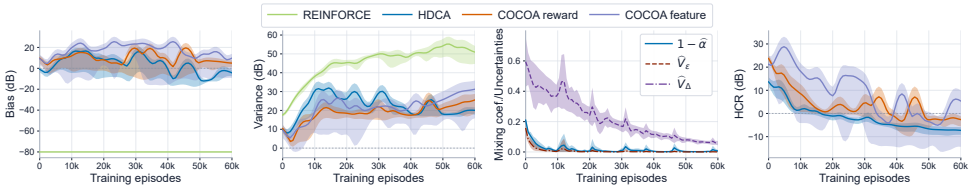


Figure 5: **Dynamics of the adaptive mixing on the `Colored-keys` task.** Bias, variance, mixing components, and HCR throughout training illustrate how HDCA initially retains more of the low-bias REINFORCE signal and progressively shifts toward the learned causal estimator as its reliability improves.

rewards. HDCA is comparatively lightweight: it does not require learning a full environment model or generating imagined rollouts. Instead, it learns only the hindsight quantities needed to estimate the contribution of an earlier action to a later outcome, and combines this learned attribution with a model-free temporal estimator.

The reliability-adaptive component of HDCA also connects to work on predictive uncertainty and uncertainty-weighted combinations of learned estimators. Deep ensembles provide a simple and effective way to quantify predictive uncertainty using disagreement among independently trained models (Lakshminarayanan et al., 2017). HDCA adopts this general principle for hindsight prediction: ensemble disagreement provides information about epistemic uncertainty, while the dispersion of the ensemble-mean action posterior reflects residual ambiguity in the source action. These quantities determine how much the causal estimator should be trusted locally. A particularly relevant analogy is Stochastic Ensemble Value Expansion (STEVE) (Buckman et al., 2018). STEVE combines model-free estimates with predictions obtained from model rollouts of different lengths and uses ensemble uncertainty to determine how much each rollout horizon should contribute. Its goal is to prevent inaccurate learned dynamics from degrading a value target. HDCA follows a related principle, namely that learned predictions should receive less weight when they are unreliable. However, it applies this principle to a different object: the uncertainty is in a learned hindsight contribution model, and the interpolation is between causal and temporal credit at individual source–outcome pairs rather than between model-based rollout horizons.

B Further Results and Analysis

B.1 The adaptive mixing balances bias and variance throughout training

Figure 5 illustrates the bias–variance trade-off induced by the adaptive mixing on the `Colored-keys` task. Throughout training, HDCA maintains lower gradient bias than both COCOA-reward and COCOA-feature, with COCOA-feature exhibiting the largest bias, plausibly reflecting the additional estimation error introduced by learning its outcome features. Early in training, however, this reduction in bias comes at the cost of higher variance relative to the causal estimators. While the learned causal contribution is still uncertain, the mixing coefficient retains a REINFORCE component, keeping the update a mixture of two estimators rather than fully trusting an inaccurate causal signal.

This behavior is governed by the evolution of the estimated $\widehat{V}_\Delta$ and $\widehat{V}_\varepsilon$, whose relative magnitudes determine how much each estimator is trusted as training progresses. Importantly, the resulting gradient trade-off is accompanied by a steady improvement in credit-assignment accuracy: the HCR of HDCA falls rapidly and remains below both COCOA variants, indicating that its estimated causal contributions at critical states are more accurate. This combination of controlled gradient bias and increasingly accurate credit assignment is consistent with the sustained performance gap in favor of HDCA on `Colored-keys` in Figure 3. Figure 5 also shows a qualitative positive association between epistemic disagreement $\widehat{V}_\Delta$ and HCR: higher credit-reconstruction error early in training accompanies larger disagreement, and both generally decrease as training progresses. Numerically, the Pearson correlation between the two quantities is 0.87 ± 0.03 during training. This correspondence suggests that disagreement may serve as an empirical proxy for hindsight-model reliability in the evaluated tasks. The results for the other tasks are in Appendix B.6. Additionally, the ground-truth quantities introduced in Appendix H enable us to compute the exact values of V_Δ and V_ε, and

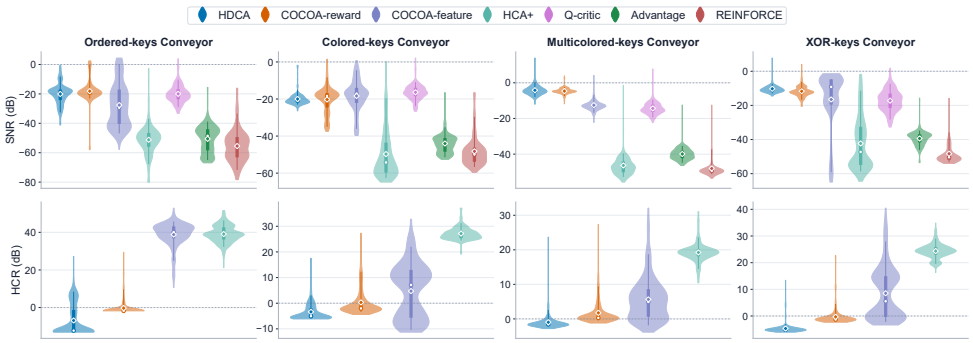


Figure 6: **Policy-gradient SNR and credit-assignment accuracy.** HDCA consistently achieves the lowest HCR across tasks while maintaining a high policy-gradient SNR comparable to the strongest causal baselines, providing both accurate credit assignment and a strong learning signal.

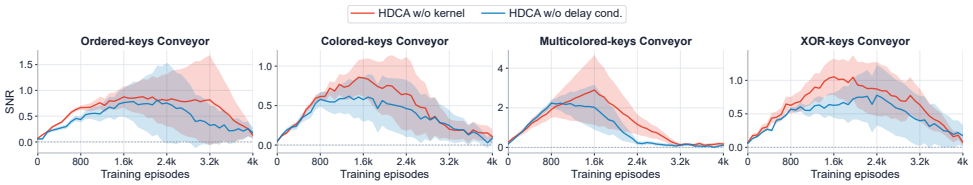


Figure 7: **Effect of delay conditioning on policy-gradient SNR.** Removing kernel smoothing isolates delay conditioning, which yields a consistently stronger SNR than pooling over delays during the early stage of training, when the learning signal is strongest.

consequently $\alpha^{\star}_{t,z}$. For a comparison among HDCA, the optimal mixing coefficient, and exponential annealing mixing, please refer to Appendix B.5.

B.2 The SNR and HCR are more favorable in HDCA

Figure 6 shows that HDCA achieves the most favorable combination of credit-assignment accuracy and training-signal strength (SNR). Across all four tasks it has the lowest HCR (lower is better), indicating it assigns credit at critical states most accurately, while maintaining an SNR comparable to COCOA-reward and among the highest evaluated. The ranking of methods by measured SNR does not consistently match their learning performance in Figure 3. For example, Q-critic exhibits relatively high SNR on `Colored-keys` despite limited learning progress, while COCOA-feature retains relatively high SNR on `XOR-keys` despite its performance decline. These observations motivate examining SNR alongside HCR when interpreting learning behavior. Across the evaluated tasks, HDCA combines low HCR with competitive SNR, accompanying its strong learning performance.

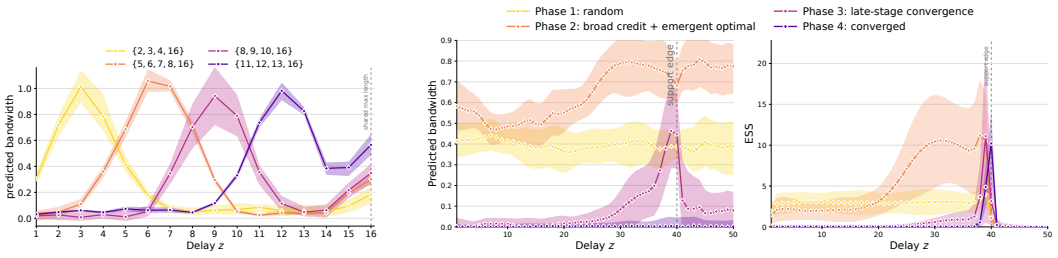
B.3 Delay conditioning leads to higher SNR

Figure 7 isolates the effect of conditioning on the source–outcome delay by comparing HDCA without kernel smoothing, which retains delay conditioning, against HDCA without delay conditioning in which $\widehat{w}_{t,z}$ is replaced by $\widehat{w}_t$ in Equation (5). Across the four tasks, delay conditioning yields a consistently stronger SNR during the early stages of training, particularly around the period in which the SNR peaks and the policy receives its strongest learning signal. This observation is consistent with the motivation of Theorem 3.2.1. Pooling across delays can dilute an action-dependent contribution by mixing informative and uninformative delays, whereas conditioning on the delay preserves the temporal specificity of the contribution and makes it easier to resolve from finite on-policy data. The resulting stronger gradient signal during the most active phase of learning supports delay conditioning as an important component of HDCA's sample-efficient credit assignment.

B.4 KERNEL SMOOTHING IS USED WHEN USEFUL AND OTHERWISE REVERTS TO THE DELAY-CONDITIONED REGIME

We first study the learned bandwidth in the `Multi-Length Conveyor`, where the initial action selects one of four conveyors that all lead to the same treasure reward but at different delays. We consider four variants in which three conveyor lengths form a nearby cluster while a fourth has the shared length (16); as in the previous experiments, ($\gamma = 1$), so shorter delays are not intrinsically preferred. Midway through training, we query the bandwidth network at the initial state and treasure outcome over all possible delays. Figure 8a shows that the learned bandwidth closely follows the local delay structure: its maximum shifts with the cluster of nearby conveyor lengths, from approximately (z=3) for (2,3,4,16) to approximately (z=12) for (11,12,13,16). Moreover, at the common delay (z=16), the bandwidth becomes larger as the remaining supported delays move closer to it. Thus, the bandwidth is not determined by the absolute delay alone; it adapts to the local support and borrows statistical strength more strongly when useful neighboring delays are available.

Figure 8b examines this adaptation over training in the `Sticky Conveyor`, where collecting an apple delays progress by one step and therefore allows the same treasure outcome to occur over a range of delays. Early in training, the bandwidth is relatively diffuse. As the agent begins collecting different numbers of apples, the treasure is observed over a broad range of delays, and the network learns a broad, nonzero bandwidth; the corresponding increase in effective sample size shows that observations are being shared across neighboring delays. Later, as the policy increasingly collects most apples, the treasure-delay distribution concentrates toward larger delays and the learned bandwidth becomes correspondingly localized around this region. At convergence, the bandwidth collapses to nearly zero over almost the entire delay range, apart from a narrow region near the support edge, empirically approaching the delay-conditioned endpoint ($\sigma \to 0$). Together, these phases show that HDCA uses smoothing when cross-delay sharing is beneficial under finite data, but progressively relinquishes it as the relevant delay becomes well resolved.



(a) (b)

Figure 8: **Adaptation of the learned kernel bandwidth.** (a) In the Multi-Length Conveyor, the predicted bandwidth adapts to the local structure of supported reward delays, increasing cross-delay sharing when nearby informative delays are available. (b) In the Sticky Conveyor, the learned bandwidth and resulting effective sample size adapt to the evolving reward-delay distribution during training and approach the delay-conditioned regime at convergence.

B.5 HDCA ADAPTIVE MIXING COMPARISON WITH OPTIMAL AND ANNEALING MIXING

In this section, we aim to compare two additional variants of convex mixing with our practical input-dependent mixing. Specifically, we consider two new variants of HDCA, referred to as HDCA-optimal-mixing and HDCA-annealing. As discussed in Appendix H, we can construct the ground-truth values of the contribution coefficients and policy gradients through dynamic programming on the conveyor family of tasks considered in our experiments. With this tool, we can further obtain the ground-truth values of V_Δ and V_ε, and consequently compute the optimal mixing coefficient $\alpha^\star_{t,z}$ as follows.

For each input $x_{t,z} = (t, s, u, z)$, the delay-conditioned posterior follows from the coefficients of Appendix H,

$$h^\star_{t,z}(a \mid s, u) = \frac{\pi_\theta(a \mid s)\, p_{t,z}(u \mid s, a)}{p_{t,z}(u \mid s)} = \pi_\theta(a \mid s)\big(1 + w^\star_{t,z}(s, a, u)\big). \tag{12}$$

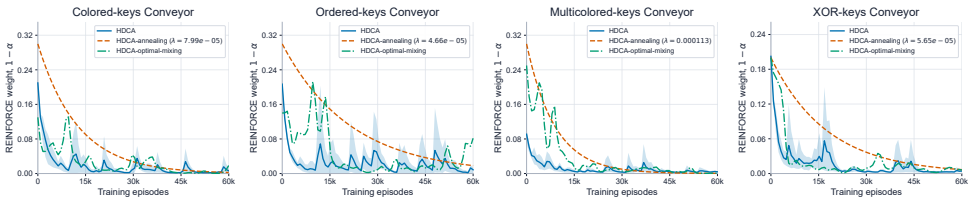


Figure 9: **Adaptive, oracle, and annealed mixing.** REINFORCE weight $(1 - \alpha)$ during training on four conveyor tasks. HDCA and the oracle mixing generally reduce reliance on temporal credit as training progresses. Exponential schedules fitted to the oracle mixes require different decay rates across tasks.

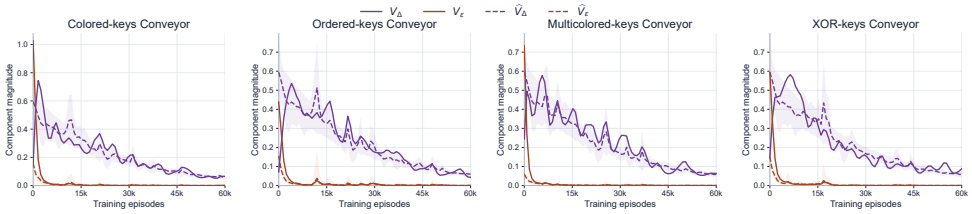


Figure 10: **Estimated and ground-truth mixing components.** Comparison of V_Δ and V_ε with their practical estimates $\widehat{V}_\Delta$ and $\widehat{V}_\varepsilon$ across four conveyor tasks. The estimates broadly track the evolution of the ground-truth components used to determine the oracle mixing weight.

Since all policies here are tabular softmax, Γ in Equation (42) acts as the identity on $\mathbf{1}^\perp$ (Equation (54) holds exactly with $c = 1$). Evaluating Equation (47) and $\|\Delta_{t,z}\|^2$, with $\Delta_{t,z} = R_{t+z} \sum_a g_a(\bar{h}_\phi - h^\star_{t,z})(a)$, at $(S_t, U_{t+z}) = (s, u)$, where $R_{t+z} = u$ under $U = R$, gives

$$V_\varepsilon(x_{t,z}) = u^2\big(1 - \big\|h^\star_{t,z}(\cdot \mid s, u)\big\|_2^2\big), \qquad V_\Delta(x_{t,z}) = u^2 \, \big\|\bar{h}_\phi(\cdot \mid x_{t,z}) - h^\star_{t,z}(\cdot \mid s, u)\big\|_2^2. \quad (13)$$

These are the pair-local conditional versions of $V_\varepsilon^{(t,z)}$ and $V_\Delta^{(t,z)}$ in Assumption E.5.3. Averaging them over S_t and $U_{t+z} \sim p_{t,z}(\cdot \mid S_t)$ recovers the population moments. Substituting Equation (13) into Equation (49) gives the oracle mixing

$$\alpha^\star_{t,z}(s, u) = \frac{V_\varepsilon(x_{t,z})/\,|\mathcal{B}|}{V_\varepsilon(x_{t,z})/\,|\mathcal{B}| + V_\Delta(x_{t,z})}, \quad (14)$$

with $\alpha^\star_{t,z} = 1$ when both terms vanish. Here $|\mathcal{B}|$ is the policy-gradient batch size (the batch $\mathcal{B}$ of Algorithm 1). The temporal residual $\varepsilon_{t,z}$ averages down over the batch, whereas $\Delta_{t,z}$ is fixed at a given input. Setting $|\mathcal{B}| = 1$ recovers Equation (49) exactly. HDCA-optimal-mixing replaces Equation (56) with Equation (14) and recomputes the mixing coefficient after every hindsight-model update.

After obtaining the optimal mixing coefficient $\alpha^\star_{t,z}$, we further fit an annealing exponential function with respect to the training episodes to the optimal mixing values. The resulting fitted curve, the optimal mixing, and our practical mixing function are all shown in Figure 9. As illustrated, the overall behavior of the optimal mixing and our practical mixing function is similar. To further examine the fine-grained signals contributing to the final optimal mixing coefficients, we also plot and compare the estimated and ground-truth values of V_Δ and V_ε in Figure 10.

In terms of performance, we compare the three variants across the four conveyor-based tasks shown in Figure 11. The performance of the three variants is relatively similar, with the optimal mixing exhibiting a faster increase in performance and both the optimal mixing and HDCA achieving higher final performance than the exponential annealing variant.

The key observation is that, although the performance of the exponential annealing variant is relatively close, it may appear that the adaptive input-dependent mixing in HDCA could be replaced by a fixed annealing schedule while maintaining similar performance. However, as shown in Figure 9, the decay rate of the exponential annealing schedule varies across tasks. This indicates that the annealing rate would need to be tuned as a task-specific hyperparameter, since the optimal $\alpha^\star$ is not available a priori. Therefore, the main advantage of HDCA's adaptive mixing mechanism is that it can achieve near-optimal mixing behavior through a hyperparameter-free procedure, eliminating the

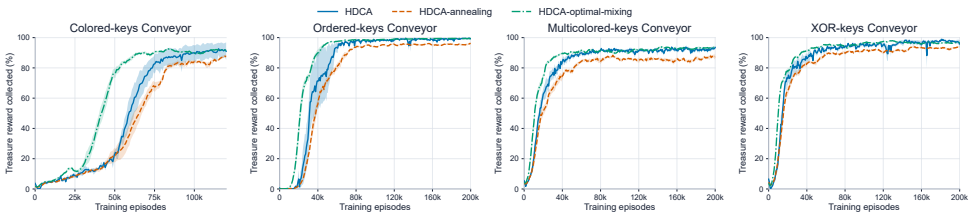


Figure 11: **Effect of the mixing strategy on learning.** Treasure reward collected by HDCA, HDCA-optimal-mixing, and HDCA-annealing across four conveyor tasks. HDCA achieves final performance comparable to oracle mixing and higher than the fitted annealing schedules, although oracle mixing accelerates learning on some tasks.

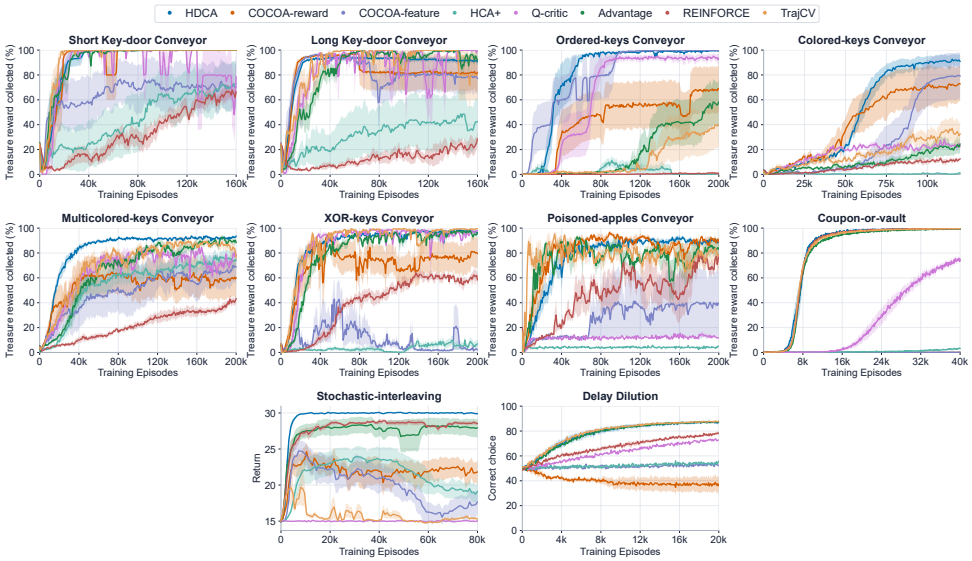


Figure 12: **Performance across the full suite of credit-assignment benchmarks.** Learning curves compare HDCA with all baselines on all conveyor-based tasks and `Stochastic-interleaving`. HDCA achieves the strongest or competitive final performance across the suite, including the maximum return on `Stochastic-interleaving`, while remaining competitive with successful temporal-credit methods on the diagnostic tasks designed to expose failure modes of purely causal credit assignment. Shaded regions denote 95% bootstrap intervals over five seeds.

need to tune the mixing mechanism for each new task and reducing the number of hyperparameters requiring careful adjustment.

B.6 Extensive Results

In this part, we extend Figure 3 by reporting the performance of our method and the baselines across the full suite of conveyor-based benchmarks and a contextual bandit one. Specifically, we include additional results for `Coupon-or-vault`, `Long Key-door Conveyor`, and `Delay Dilution`, as well as the contextual bandit task originally proposed in Meulemans et al. (2023), `Stochastic-interleaving`.

The additional benchmarks in Figure 12 further reinforce the pattern observed in Section 4.3, showing that our method is the only approach that consistently achieves the highest performance across all tasks. In particular, on `Stochastic-interleaving`, our method reaches the maximum possible return, resulting in a persistent performance gap relative to the baselines. Moreover, on the two diagnostic tasks designed to highlight the limitations of purely causal credit-assignment methods, our method operates in the same performance regime as successful temporal credit-assignment approaches.

Additionally, we extend the analysis of the adaptive mixing coefficient in Appendix B.1 beyond the `Colored-keys Conveyor` task (Figure 13). The same qualitative behavior appears on both

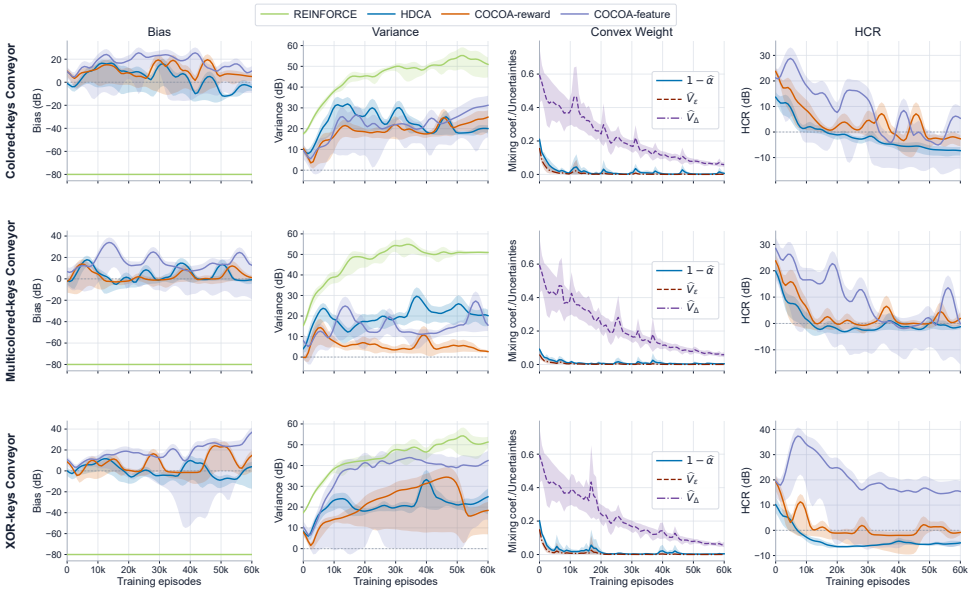


Figure 13: **Adaptive-mixing coefficient dynamics across additional credit-assignment tasks.** Rows show the `Colored-keys`, `Multicolored-keys`, and `XOR-keys` Conveyor tasks, while columns report policy-gradient bias, variance, the estimated mixing coefficient components and resulting convex weight, and HCR throughout training. REINFORCE provides the low-bias but high-variance temporal reference, whereas the causal estimators can reduce variance at the cost of bias when their learned contribution coefficients are inaccurate. HDCA adaptively interpolates between these regimes, controlling gradient bias while generally maintaining more accurate credit reconstruction. The decreasing ensemble disagreement illustrates how the estimated reliability of the causal estimator evolves during training.

`Multicolored-keys Conveyor` and `XOR-keys Conveyor`: HDCA remains substantially less biased than the fully causal estimators while avoiding the much larger variance of the temporal estimator. As training progresses, the estimated uncertainty terms adjust the convex weight according to the reliability of the learned causal contribution, allowing HDCA to interpolate between the two estimators rather than committing to either one throughout training. This trade-off is accompanied by consistently favorable HCR, indicating that the reduction in gradient error does not come at the expense of credit-assignment accuracy. Together, these results suggest that the behavior observed on `Colored-keys Conveyor` is not task-specific, but reflects a broader property of the reliability-adaptive mixing mechanism.

C Ablation Study

C.1 Ablated Variants

We evaluate the contributions of delay conditioning, kernel smoothing, and adaptive mixing in HDCA. Figure 14 compares full HDCA with five variants: without delay conditioning, without kernel smoothing, with fixed $\alpha = 0.5$, and COCOA-reward with delay conditioning, either alone or together with kernel smoothing. The $\alpha = 0.5$ variant assigns equal weight to the temporal and causal estimators while retaining delay conditioning and kernel smoothing. The two COCOA-reward variants correspond to fixing $\alpha = 1$ in HDCA and setting $M = 1$, which replaces the hybrid estimator with the causal estimator alone. Since HDCA uses the scalar reward as its outcome representation, this endpoint is exactly delay-conditioned COCOA-reward, with or without additional kernel smoothing.

The benefits of delay conditioning and kernel smoothing depend on the task. Removing delay conditioning substantially delays learning in `Ordered-keys` and slows progress in `Colored-keys`. Removing kernel smoothing also delays learning in `Ordered-keys` and leads to a pronounced late-training performance decline in `Poisoned-apples`. The effects of these two ablations are comparatively smaller in `Multicolored-keys` and `XOR-keys`.

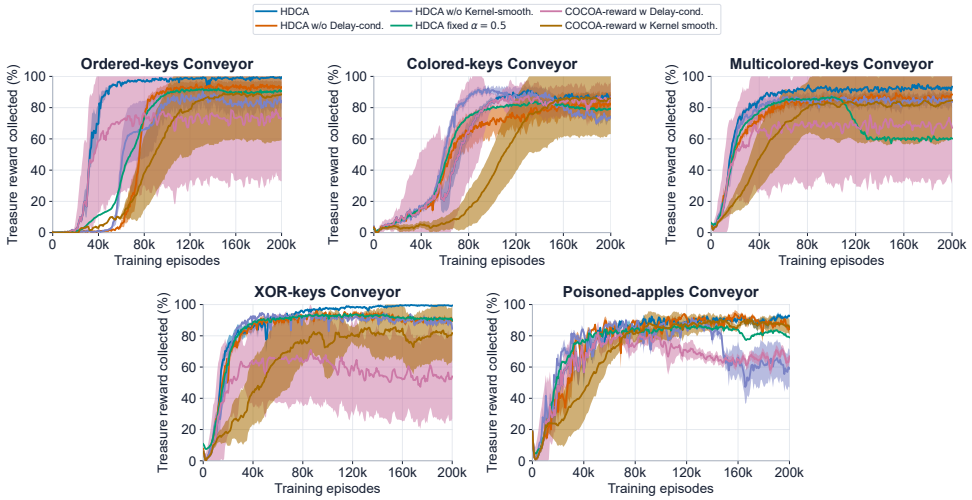


Figure 14: **Ablation of adaptive mixing, delay conditioning, and kernel smoothing.** Learning curves compare full HDCA with five variants: without delay conditioning, without kernel smoothing, fixed $\alpha = 0.5$, and COCOA-reward with delay conditioning, alone or with additional kernel smoothing. Fixed $\alpha = 0.5$ equally weights the temporal and causal estimators; the COCOA-reward variants correspond to fixing $\alpha = 1$, which uses only the causal estimator. Removing delay conditioning or kernel smoothing delays learning on `Ordered-keys`, and removing smoothing causes a late-training decline on `Poisoned-apples`. Equal mixing exhibits a pronounced decline on `Multicolored-keys`. Delay-conditioned COCOA-reward without smoothing has lower sustained performance on several tasks, including declines on `XOR-keys` and `Poisoned-apples`. Kernel smoothing improves its eventual performance on several tasks, but substantial learning delays remain. These results illustrate the complementary contributions of the three components to sample efficiency and training stability.

Replacing the adaptive mixing with fixed $\alpha = 0.5$ substantially slows learning in `Ordered-keys`. On `Multicolored-keys`, equal mixing initially reaches high success but subsequently declines to approximately 60%, while full HDCA maintains success above 90%. Differences are smaller on the remaining tasks. These results suggest that an adaptive mixing mechanism based on the reliability of the learned causal contribution can improve both sample efficiency and training stability relative to equal mixing.

The COCOA-reward variants further clarify the contribution of adaptive mixing. Without kernel smoothing, delay-conditioned COCOA-reward improves rapidly early in training on some tasks, but plateaus below full HDCA on `Ordered-keys` and `Multicolored-keys`, and exhibits late-training declines on `XOR-keys` and `Poisoned-apples`. Its uncertainty bands are also particularly wide on several tasks. Adding kernel smoothing improves its eventual performance on `Ordered-keys`, `Multicolored-keys`, `XOR-keys`, and `Poisoned-apples`, but learning remains slower than with full HDCA, most clearly on `Colored-keys`. Together, these comparisons support complementary, task-dependent benefits from delay conditioning, kernel smoothing, and adaptive weighting of temporal and causal credit.

C.2 Different Ensemble Sizes

HDCA uses an ensemble of M hindsight models both to estimate contribution coefficients (through the ensemble mean) and to measure their reliability (through ensemble disagreement, which sets the mixing coefficient $\widehat{\alpha}_{t,z}$ in Section 3). To test whether its gains simply come from averaging several hindsight models, we vary $M \in \{2, \ldots, 8\}$ and compare HDCA with an ensembled COCOA-reward baseline that uses the mean of M hindsight models but no adaptive mixing (Figure 15). HDCA's performance increases with M and approaches 100% on all four tasks, with most of the gain obtained by $M = 5$, the value used in our main experiments. In contrast, ensembling improves COCOA-reward only marginally on Colored-keys, Ordered-keys, and Multicolored-keys, where it stays roughly 20-30 percentage points below HDCA at every ensemble size. On XOR-keys, both methods perform well, and COCOA-reward is slightly better at small M before both converge. These results suggest that the benefit of HDCA does not come from ensembling alone: larger ensembles help mainly

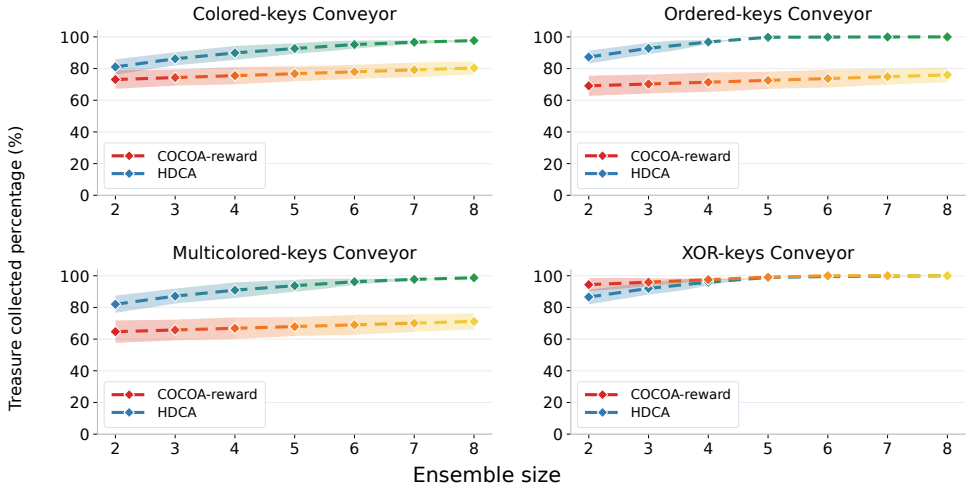


Figure 15: **Effect of ensemble size.** Best treasure-collection rate (throughout training evaluated every 100 episodes) of HDCA and an ensembled COCOA-reward baseline as the number of hindsight models M grows from 2 to 8, on four conveyor tasks (shaded regions: variability across seeds). HDCA improves steadily with M and reaches near-optimal performance on all tasks, whereas COCOA-reward gains only slightly, indicating that HDCA's advantage comes from using ensemble disagreement to mix causal credit rather than from ensembling itself.

because they give a more reliable disagreement signal for deciding how much to trust the causal estimator.

D Definitions, Assumptions, Conventions, and Notation

Throughout the appendix we use the notation fixed in Table 1. The symbol g always denotes the true policy gradient $\nabla_\theta J(\theta)$; a generic candidate predictor in a log-loss argument is written f; the hindsight posterior is h and the kernel bandwidth is σ; ensemble disagreement is $\mathcal{U}$ and the outcome encoding is U. Results here are stated for a general discount $\gamma \in (0, 1]$; the main text specializes them to $\gamma = 1$, where the temporal estimator is unbiased and the term $b_R(\gamma)$ below vanishes. Whenever a result below uses a general γ, its hybrid is defined by

$$\widehat{G}_H^\gamma := \Pi + \sum_{(t,z)\in\mathcal{I}_T} \left[\alpha_{t,z}\widehat{G}_{CD}^{(t,z)} + (1-\alpha_{t,z})G_R^{\gamma,(t,z)}\right].$$

We suppress this superscript in the general-γ analysis. At $\gamma = 1$ this is exactly Equation (6).

Here, we first develop the notation from the beginning and provide a more detailed treatment of the notation, derivations, and assumptions that were presented in a compressed form earlier. The RL objective is

$$J(\theta) := \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} R_t\right], \qquad g := \nabla_\theta J(\theta), \tag{15}$$

and the policy is assumed to have full support on every enumerated state–action pair,

$$\pi_\theta(a \mid s) > 0 \qquad \text{for every relevant } (s, a), \tag{16}$$

which is what makes all hindsight and score ratios well defined. The policy-averaged and uniformly pooled outcome likelihoods are

$$p_{t,z}(u \mid s) := \sum_{b\in\mathcal{A}} \pi_\theta(b \mid s)p_{t,z}(u \mid s, b), \qquad \bar{p}_t(u \mid s, a) := \frac{1}{H_t}\sum_{z=1}^{H_t} p_{t,z}(u \mid s, a), \tag{17}$$

and $\bar{p}_t(u \mid s)$ is defined analogously. The pair index set used for per-pair mixing is

$$\mathcal{I}_T := \{(t, z) : 0 \le t \le T-2,\ 1 \le z \le H_t\}, \qquad N_T := |\mathcal{I}_T| = \frac{T(T-1)}{2}. \tag{18}$$

The delay share and the averaging identity relating the pooled and delay-conditioned coefficients are

$$\rho_{t,z}(s,u) := \frac{p_{t,z}(u \mid s)}{\sum_{j=1}^{H_t} p_{t,j}(u \mid s)}, \qquad w_t^\star(s,a,u) = \sum_{z=1}^{H_t} \rho_{t,z}(s,u)\, w_{t,z}^\star(s,a,u). \tag{19}$$

Table 1: Notation. Each symbol is used with this meaning.

Symbol	Meaning
g	true policy gradient $\nabla_\theta J(\theta)$ (never a predictor or a direction)
f	generic candidate predictor in log-loss arguments
$h_t^\star,\ h_{t,z}^\star, \widehat{h}_t,\ \widehat{h}_{t,z},\ \bar{h}_\phi$	pooled, delay-conditioned (optimal and learned), and learned ensemble-mean hindsight posteriors
σ	kernel bandwidth; ω denotes normalized kernel weights
$U,\ u$	rewarding-outcome encoding; $\mathcal{U}$ is ensemble disagreement
$Z,\ z$	action-to-outcome delay; $H_t = T - 1 - t$ is the number of available delays
$\gamma,\ \gamma_b$	policy discount factor; kernel bias exponent
D_z	nuisance (distractor) reward component at delay z
$G_R^\gamma, G_C^\star, \widehat{G}_C, G_{CD}^\star, \widehat{G}_{CD}, \widehat{G}_H$	temporal; pooled oracle and learned pooled causal; delay-conditioned oracle and learned delay-conditioned causal; and hybrid estimators
$G_C^{\star,(t,z)},\ G_{CD}^{\star,(t,z)}$	pairwise oracle causal terms built from the *pooled* coefficient $w_t^\star$ and from the *delay-conditioned* coefficient $w_{t,z}^\star$
$\|v\|_{\bullet,\mathcal{F}}$	$:= \max_{(t,z)\in\mathcal{I}_T} \left\{\mathbb{E}[v_{t,z}^2 \mid \mathcal{F}]\right\}^{1/2}$ for a random vector indexed by $\mathcal{I}_T$
Λ_C, Λ_{CD}	local (single-source, single-action) future contributions
$\alpha,\ \alpha_{t,z},\ \alpha_{t,z}^\star,\ \widehat{\alpha}_{t,z}$	generic constant mixing, generic per-pair causal mixing, oracle MSE-optimal mixing, and practical adaptive mixing used by HDCA, respectively; all mixing coefficients lie in $[0,1]$
$B, \Delta, \varepsilon_{t,z}$	structural projection gap, causal-estimator discrepancy/error, and temporal residual, as defined locally in each analysis
C_G	finite bound on the summed conditional pairwise estimator discrepancy
$\sigma(X)$	sigma-field generated by the random variable, or collection of random variables, X
$\mathcal{F}_n$	sigma-field generated by the complete training history before the iteration-n policy-gradient batch is sampled
$\mathcal{H}$	projection sigma-field used in the adaptive-mixing analysis
$\mathcal{H}_{t,z}$	pair-local information set $\sigma(S_t, U_{t+z})$ (distinct from the global $\mathcal{H}$ of Appendix E.4)
$\varepsilon_{h,n}$	average on-policy ℓ_1 hindsight-posterior error
$\rho_{t,z}$	share of the outcome's occurrence probability carried by delay z
$\mathcal{V}(X)$	shows the variance of random variable X
$\mathcal{C}(X,Y)$	shows the covariance of random variables X and Y
$x \downarrow k$	means x decreases monotonically toward k

Definition D.0.1 (Total variance and covariance)**.** For square-integrable random vectors X, Y,

$$\mathcal{V}(X) := \mathbb{E}\,\|X - \mathbb{E}X\|^2 = \operatorname{tr}\operatorname{Cov}(X), \tag{20}$$

$$\mathcal{C}(X,Y) := \mathbb{E}\,\langle X - \mathbb{E}X, Y - \mathbb{E}Y\rangle = \operatorname{tr}\operatorname{Cov}(X,Y). \tag{21}$$

Matrix inequalities use the positive-semidefinite order $A \preceq B$, meaning $\mathbf{v}^\top A\mathbf{v} \le \mathbf{v}^\top B\mathbf{v}$ for every $\mathbf{v} \in \mathbb{R}^d$.

Definition D.0.2 (Information sets)**.** A sigma-field is a collection of events closed under complements and countable unions; it represents the information known at a particular stage of the exper-

iment. Let $\mathcal{F}_n$ denote the sigma-field generated by the complete training history before the policy-gradient batch at iteration n is sampled. Conditional on $\mathcal{F}_n$, the current policy parameters and all previously trained model parameters are fixed, while the current batch remains random. A random variable is $\mathcal{F}_n$-measurable if its value is determined once this history is known.

Assumption D.0.3 (Policy-gradient regularity)**.** *The initial-state distribution and the environment transition–reward law do not depend directly on θ. The policy is differentiable and has full support (Equation* (16)*), differentiation may be interchanged with trajectory expectation in Equation* (15)*, and every gradient estimator appearing in a variance or MSE statement is square-integrable, $\mathbb{E}\,\|G\|^2 < \infty$.*

Assumption D.0.4 (Bounded rewards and policy scores)**.** *There exist constants $R_{\max}, G_{\max} < \infty$ such that $|R_t| \le R_{\max}$ and $\|\nabla_\theta \log \pi_\theta(a \mid s)\| \le G_{\max}$ for every time and every state–action pair in the relevant on-policy support.*

Assumption D.0.5 (Bounded pairwise estimator discrepancy)**.** *For every pair $(t, z) \in \mathcal{I}_T$ and every iteration n there is a finite constant $C_{t,z}$, not depending on n, with*

$$\Big(\mathbb{E}\big[\Big\|\widehat{G}_{CD}^{(t,z)} - G_R^{\gamma,(t,z)}\Big\|^2 \mid \mathcal{F}_n\big]\Big)^{1/2} \le C_{t,z}, \qquad C_G := \sum_{(t,z)\in\mathcal{I}_T} C_{t,z} < \infty. \tag{22}$$

Under Assumption D.0.4 and a uniform bound on the learned coefficients this holds with $C_G = O(T^2)$. Because C_G grows quadratically in the horizon, the bias bounds of Appendix E.9 are informative only in the regimes they are designed for. They are useful when the relevant mixing norm is small relative to C_G^{-1}, and are otherwise dominated by the trivial bound obtained from Assumption D.0.4. They are qualitative statements about which estimator anchors the bias, not sharp constants.

E Hybrid Gradient Estimator

E.1 Discounting bias of the temporal estimator

Proposition E.1.1 (Exact discounting bias)**.** *Under Assumption D.0.3, $\mathbb{E}[G_R^\gamma] = g + b_R(\gamma)$ with*

$$b_R(\gamma) = -\sum_{t=0}^{T-2}\sum_{z=1}^{H_t}(1-\gamma^z)\,\mathbb{E}\big[\nabla_\theta \log \pi_\theta(A_t \mid S_t) R_{t+z}\big], \tag{23}$$

and $b_R(1) = 0$. Under Assumption D.0.4,

$$\|b_R(\gamma)\| \le G_{\max} R_{\max}\,(1-\gamma)\,\frac{T(T-1)(T+1)}{6} \le \tfrac{1}{6} G_{\max} R_{\max} T^3 (1-\gamma). \tag{24}$$

Proof. Write $\ell_t := \nabla_\theta \log \pi_\theta(A_t \mid S_t)$. Because the initial-state and environment laws do not depend on θ, Assumption D.0.3 gives the likelihood-ratio identity

$$g = \mathbb{E}\left[\left(\sum_{t=0}^{T-1} \ell_t\right)\left(\sum_{k=0}^{T-1} R_k\right)\right].$$

Let $\mathcal{P}_t$ contain the trajectory history up to S_t, before A_t is sampled. Conditional on this history,

$$\mathbb{E}[\ell_t \mid \mathcal{P}_t] = \sum_a \pi_\theta(a \mid S_t)\nabla_\theta \log \pi_\theta(a \mid S_t) = \nabla_\theta \sum_a \pi_\theta(a \mid S_t) = 0.$$

For $k < t$, R_k is $\mathcal{P}_t$-measurable, so $\mathbb{E}[\ell_t R_k] = \mathbb{E}[R_k \mathbb{E}[\ell_t \mid \mathcal{P}_t]] = 0$. Consequently,

$$g = \sum_{t=0}^{T-1}\sum_{k=t}^{T-1} \mathbb{E}[\ell_t R_k] = \mathbb{E}[G_R^1].$$

By Equation (3), discounting multiplies each future pair by γ^{k-t} and leaves the immediate term unchanged. Subtracting the preceding identity therefore gives

$$\mathbb{E}[G_R^\gamma] - g = \sum_{t=0}^{T-2}\sum_{z=1}^{H_t}(\gamma^z - 1)\mathbb{E}[\ell_t R_{t+z}] = b_R(\gamma),$$

which proves Equation (23) and $b_R(1) = 0$. Under Assumption D.0.4, $\|\mathbb{E}[\ell_t R_{t+z}]\| \le \mathbb{E}[\|\ell_t\|\,|R_{t+z}|] \le G_{\max} R_{\max}$. Also $1-\gamma^z = (1-\gamma)\sum_{j=0}^{z-1}\gamma^j \le z(1-\gamma)$. Thus

$$\begin{aligned}\|b_R(\gamma)\| &\le G_{\max} R_{\max}(1-\gamma)\sum_{t=0}^{T-2}\sum_{z=1}^{H_t} z \\ &= G_{\max} R_{\max}(1-\gamma)\sum_{j=1}^{T-1}\frac{j(j+1)}{2} \\ &= G_{\max} R_{\max}(1-\gamma)\frac{T(T-1)(T+1)}{6} \le \frac{1}{6} G_{\max} R_{\max} T^3(1-\gamma),\end{aligned}$$

as required. □

E.2 Pooled hindsight identities

Lemma E.2.1 (Bayes form of the pooled coefficient)**.** *On the support where $\bar{p}_t(u \mid s) > 0$ and $\pi_\theta(a \mid s) > 0$,*

$$\frac{h^\star_t(a \mid s,u)}{\pi_\theta(a \mid s)} = \frac{\bar{p}_t(u \mid s,a)}{\bar{p}_t(u \mid s)} = \frac{\sum_{z=1}^{H_t} p_{t,z}(u \mid s,a)}{\sum_{z=1}^{H_t} p_{t,z}(u \mid s)}. \tag{25}$$

Proof. Because K_t is uniform on $\{t+1,\dots,T-1\}$ and, given the source time, independent of the trajectory, $\mathbb{P}(U_{K_t} = u \mid S_t = s, A_t = a) = H_t^{-1}\sum_z p_{t,z}(u \mid s,a) = \bar{p}_t(u \mid s,a)$, and likewise without conditioning on A_t. Bayes' rule gives $h^\star_t(a \mid s,u) = \pi_\theta(a \mid s)\bar{p}_t(u \mid s,a)/\bar{p}_t(u \mid s)$. Dividing by $\pi_\theta(a \mid s)$ and cancelling the common factor H_t^{-1} in numerator and denominator proves Equation (25). □

Lemma E.2.2 (The pooled coefficient is a ρ-weighted average)**.** *With $\rho_{t,z}(s,u)$ as in Equation* (19)*, $w^\star_t(s,a,u) = \sum_{z=1}^{H_t}\rho_{t,z}(s,u)\, w^\star_{t,z}(s,a,u)$.*

Proof. Using Equation (25) and $p_{t,z}(u \mid s,a) = p_{t,z}(u \mid s)\{1 + w^\star_{t,z}(s,a,u)\}$,

$$w^\star_t = \frac{\sum_z p_{t,z}(u \mid s)\{1 + w^\star_{t,z}\}}{\sum_j p_{t,j}(u \mid s)} - 1 = \sum_z \rho_{t,z} w^\star_{t,z}, \tag{26}$$

since the weights $\rho_{t,z}$ sum to one. □

Lemma E.2.3 (Delay posterior under the exogenous-delay convention)**.** *Let $X := (t, S_t, U_{t+Z})$ be the hindsight-model input generated by the exogenous delay convention of Section 3. Then for every supported $x = (t,s,u)$,*

$$\mathbb{P}(Z = z \mid X = x) = \rho_{t,z}(s,u) = \frac{p_{t,z}(u \mid s)}{\sum_{j=1}^{H_t} p_{t,j}(u \mid s)}. \tag{27}$$

Proof. Conditional on the source time t and $S_t = s$, the delay Z is uniform on $\{1,\dots,H_t\}$ and independent of the trajectory, so $\mathbb{P}(Z = z \mid S_t = s) = 1/H_t$ and $\mathbb{P}(U_{t+Z} = u \mid S_t = s, Z = z) = p_{t,z}(u \mid s)$. Bayes' rule gives

$$\mathbb{P}(Z = z \mid S_t = s, U_{t+Z} = u) = \frac{H_t^{-1} p_{t,z}(u \mid s)}{\sum_j H_t^{-1} p_{t,j}(u \mid s)} = \rho_{t,z}(s,u). \tag{28}$$

□

Lemma E.2.3 is used repeatedly below. It says that the very same weights ρ that attenuate the pooled coefficient in Lemma E.2.2 also determine how the hindsight training data is divided among delays. This coupling is what makes the trade-off in Theorem F.4.1 exactly computable.

E.3 UNBIASEDNESS OF THE VANILLA POOLED-DELAY ESTIMATOR

Calling U_k a rewarding-outcome representation is not by itself sufficient for unbiasedness. The pooled posterior in Equation (1) is used for rewards at every future delay, so the reward associated with a given outcome must be compatible across those delays.

Assumption E.3.1 (Pooled-delay reward sufficiency)**.** *For every source time $t \leq T-2$ there is an integrable measurable function m_t such that, for every delay $z \in \{1, \dots, H_t\}$ and almost every supported (s, a, u),*

$$\mathbb{E}[R_{t+z} \mid S_t = s, A_t = a, U_{t+z} = u] = m_t(u). \tag{29}$$

The above assumption implies that the same function m_t must apply to every delay z; this is the substantive content of the assumption. For fixed t, COCOA uses a stronger, fully predictive requirement: the conditional law of R_{t+z} given $U_{t+z} = u$ is invariant to s, a, and z. This requirement implies Equation (29) (Meulemans et al., 2023), whereas the policy-gradient result below needs only equality of the conditional means. Importantly, if the encoding is the scalar reward itself, $U_k = R_k$, then $\mathbb{E}[R_{t+z} \mid S_t, A_t, U_{t+z} = u] = u$ identically and Assumption E.3.1 holds automatically with $m_t(u) = u$. All structural-bias statements below are therefore vacuous for $U = R$ and become substantive only for encodings that do not determine the reward.

Lemma E.3.2 (Pooled future-advantage identity)**.** *Under Assumption E.3.1, for every supported (s, a, t),*

$$\sum_{z=1}^{H_t} \mathbb{E}\big[w_t^\star(s, a, U_{t+z}) R_{t+z} \mid S_t = s\big] = \sum_{z=1}^{H_t} \big\{\mathbb{E}[R_{t+z} \mid S_t = s, A_t = a] - \mathbb{E}[R_{t+z} \mid S_t = s]\big\}. \tag{30}$$

Proof. Fix (t, s, a) and abbreviate $\bar{p}_a(u) := \bar{p}_t(u \mid s, a)$, $\bar{p}(u) := \bar{p}_t(u \mid s)$, $m(u) := m_t(u)$. Averaging Equation (29) over A_t gives $\mathbb{E}[R_{t+z} \mid S_t = s, U_{t+z} = u] = m(u)$ for every z. Using Equation (25) and a common dominating measure ν[2],

$$\begin{aligned}
\sum_{z=1}^{H_t} \mathbb{E}\left[\frac{\bar{p}_a(U_{t+z})}{\bar{p}(U_{t+z})} R_{t+z} \,\middle|\, S_t = s\right] &= \int \frac{\bar{p}_a(u)}{\bar{p}(u)} m(u) \Big\{\sum_{z=1}^{H_t} p_{t,z}(u \mid s)\Big\} d\nu(u) \\
&= H_t \int \bar{p}_a(u) m(u)\, d\nu(u) = \sum_{z=1}^{H_t} \int p_{t,z}(u \mid s, a) m(u)\, d\nu(u) \\
&= \sum_{z=1}^{H_t} \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a].
\end{aligned} \tag{31}$$

Subtracting $\sum_z \mathbb{E}[R_{t+z} \mid S_t = s]$ and using $w_t^\star = \bar{p}_a / \bar{p} - 1$ proves Equation (30). □

Theorem E.3.3 (Unbiased vanilla pooled-delay causal gradient)**.** *Under Assumptions D.0.3 and E.3.1, the oracle estimator in Equation* (2) *satisfies $\mathbb{E}[G_C^\star] = \nabla_\theta J(\theta)$.*

Proof. Our proof is similar to that of Meulemans et al. (2023). Let $C_t^\star$ denote the source-time-t summand in Equation (2), so that $G_C^\star = \sum_t C_t^\star$. Define

$$Q_t^\pi(s, a) := \mathbb{E}\left[\sum_{k=t}^{T-1} R_k \,\middle|\, S_t = s, A_t = a\right].$$

Fix t and condition on $S_t = s$. Since $\mathbb{E}[\mathbf{1}\{A_t = a\} R_t \mid S_t = s] = \pi_\theta(a \mid s) \mathbb{E}[R_t \mid S_t = s, A_t = a]$, the immediate term contributes $\sum_a \nabla_\theta \pi_\theta(a \mid s) \mathbb{E}[R_t \mid S_t = s, A_t = a]$. For the future part,

[2] Writing every conditional outcome law as a density requires a common yardstick: a reference measure fixed once for all the conditional outcome laws, so they can be written as densities and combined under a single integral; take counting measure if U is discrete and Lebesgue measure if it is continuous.

Lemma E.3.2 yields

$$\begin{aligned}
&\sum_a \nabla_\theta \pi_\theta(a \mid s) \sum_{z=1}^{H_t} \mathbb{E}[w_t^\star(s,a,U_{t+z}) R_{t+z} \mid S_t = s] \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s) \sum_{z=1}^{H_t} \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a] \\
&\quad - \left(\sum_a \nabla_\theta \pi_\theta(a \mid s) \right) \sum_{z=1}^{H_t} \mathbb{E}[R_{t+z} \mid S_t = s] \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s) \sum_{z=1}^{H_t} \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a],
\end{aligned}$$

because $\sum_a \nabla_\theta \pi_\theta(a \mid s) = 0$. Adding the immediate term gives $\mathbb{E}[C_t^\star \mid S_t = s] = \sum_a \nabla_\theta \pi_\theta(a \mid s) Q_t^\pi(s,a)$. The same identity holds at $t = T-1$ with an empty future sum. Taking expectations and using $\nabla_\theta \pi_\theta = \pi_\theta \nabla_\theta \log \pi_\theta$,

$$\begin{aligned}
\mathbb{E}[G_C^\star] &= \sum_{t=0}^{T-1} \mathbb{E}\left[\sum_a \nabla_\theta \pi_\theta(a \mid S_t) Q_t^\pi(S_t, a) \right] \\
&= \sum_{t=0}^{T-1} \mathbb{E}[\nabla_\theta \log \pi_\theta(A_t \mid S_t) Q_t^\pi(S_t, A_t)] \\
&= \sum_{t=0}^{T-1} \mathbb{E}\left[\nabla_\theta \log \pi_\theta(A_t \mid S_t) \sum_{k=t}^{T-1} R_k \right] = g.
\end{aligned}$$

The penultimate equality is iterated expectation conditional on (S_t, A_t); the last is the return-to-go identity proved in Proposition E.1.1. □

E.4 Adaptive per-pair mixing: exact global projection-gap analysis

This section gives an exact global accounting of what a data-dependent per-pair mixing can do to the mean and covariance of the hybrid estimator. In particular, one cannot derive the HDCA mixing by simply declaring the causal estimator to be a Rao–Blackwellization of REINFORCE: for the summed multi-reward estimator that identity generally fails (Proposition E.4.2). The pair-local Rao–Blackwell identity that holds under $U = R$, and that is used to derive the implemented mixing, is developed separately in Appendix E.5.

Decomposing the pooled oracle estimator of Equation (2) by source–delay pair gives

$$G_C^{\star,(t,z)} := \sum_{a \in \mathcal{A}} \nabla_\theta \pi_\theta(a \mid S_t)\, w_t^\star(S_t, a, U_{t+z})\, R_{t+z}, \qquad G_C^\star = \Pi + \sum_{(t,z) \in \mathcal{I}_T} G_C^{\star,(t,z)}, \tag{32}$$

which differs from $G_{CD}^{\star,(t,z)}$ in Equation (5) only in using the pooled coefficient $w_t^\star$ in place of $w_{t,z}^\star$: the delay z enters $G_C^{\star,(t,z)}$ solely through the outcome U_{t+z} and reward R_{t+z} it multiplies.

Let $\mathcal{H}$ be a sigma-field with respect to which the shared immediate-reward term Π, the oracle pooled pairwise causal terms $G_C^{\star,(t,z)}$, the learned pairwise causal terms $\widehat{G}_{CD}^{(t,z)}$, and the mixing coefficients $\alpha_{t,z} \in [0,1]$ are all measurable. A canonical choice is

$$\mathcal{H} = \mathcal{F} \vee \sigma\big(\Pi,\ \{G_C^{\star,(t,z)}\}_{(t,z) \in \mathcal{I}_T},\ \{\widehat{G}_{CD}^{(t,z)}\}_{(t,z) \in \mathcal{I}_T},\ \alpha\big), \tag{33}$$

where $\mathcal{F}$ is the pre-batch history. Define

$$\bar{G} := \mathbb{E}[G_R^\gamma \mid \mathcal{H}], \quad \varepsilon := G_R^\gamma - \bar{G}, \quad \bar{G}_{t,z} := \mathbb{E}[G_R^{\gamma,(t,z)} \mid \mathcal{H}], \quad \varepsilon_{t,z} := G_R^{\gamma,(t,z)} - \bar{G}_{t,z}, \tag{34}$$

so $\mathbb{E}[\varepsilon \mid \mathcal{H}] = 0$ and, because Π is $\mathcal{H}$-measurable, $\bar{G} = \Pi + \sum_{t,z} \bar{G}_{t,z}$ and $\varepsilon = \sum_{t,z} \varepsilon_{t,z}$ (these residuals are local to this subsection). Define

$$B_{t,z} := G_C^{\star,(t,z)} - \bar{G}_{t,z}, \qquad \Delta_{t,z} := \widehat{G}_{CD}^{(t,z)} - G_C^{\star,(t,z)}. \tag{35}$$

Here $B_{t,z}$ is the structural projection gap: it records the failure of the pooled oracle causal term to equal the global conditional projection of its temporal counterpart. The term $\Delta_{t,z}$ records the discrepancy between the learned causal term actually inserted into the hybrid and that oracle reference. It is also deliberately different from the pair-local decomposition of Appendix E.5, where the reference causal term is the delay-conditioned oracle $G_{CD}^{\star,(t,z)}$.

Theorem E.4.1 (Exact adaptive mean and covariance with structural and estimator discrepancy)**.** *Under Assumption D.0.3, let $\widehat{G}_H$ be as in Equation* (6) *and set $\eta := \sum_{t,z}(1-\alpha_{t,z})\varepsilon_{t,z}$. Then $\widehat{G}_H = W + \eta$ with*

$$W := \bar{G} + \sum_{t,z} \alpha_{t,z}(B_{t,z} + \Delta_{t,z})$$

$\mathcal{H}$-measurable and $\mathbb{E}[\eta \mid \mathcal{H}] = 0$. Consequently,

$$\mathbb{E}[\widehat{G}_H] - \mathbb{E}[G_R^\gamma] = \mathbb{E}\Big[\sum_{t,z} \alpha_{t,z}(B_{t,z} + \Delta_{t,z})\Big], \tag{36}$$

$$\mathrm{Cov}(\widehat{G}_H) = \mathrm{Cov}\Big(\bar{G} + \sum_{t,z} \alpha_{t,z}(B_{t,z} + \Delta_{t,z})\Big) + \mathbb{E}\big[\mathrm{Cov}(\eta \mid \mathcal{H})\big]. \tag{37}$$

Therefore $\mathrm{Cov}(\widehat{G}_H) \preceq \mathrm{Cov}(G_R^\gamma)$ if and only if

$$\mathrm{Cov}\Big(\bar{G} + \sum_{t,z} \alpha_{t,z}(B_{t,z} + \Delta_{t,z})\Big) - \mathrm{Cov}(\bar{G}) \preceq \mathbb{E}\big[\mathrm{Cov}(\varepsilon \mid \mathcal{H}) - \mathrm{Cov}(\eta \mid \mathcal{H})\big]. \tag{38}$$

For a uniform mixing $\alpha_{t,z} \equiv \alpha$, we have $\eta = (1-\alpha)\varepsilon$ and the right-hand side becomes $\mathbb{E}[(2\alpha - \alpha^2)\,\mathrm{Cov}(G_R^\gamma \mid \mathcal{H})]$.

Proof. Substituting $G_R^{\gamma,(t,z)} = \bar{G}_{t,z} + \varepsilon_{t,z}$ and $\widehat{G}_{CD}^{(t,z)} = G_C^{\star,(t,z)} + \Delta_{t,z}$ into Equation (6) gives

$$\begin{aligned}\widehat{G}_H &= \Pi + \sum_{t,z}\Big[\alpha_{t,z}\big(G_C^{\star,(t,z)} + \Delta_{t,z}\big) + (1-\alpha_{t,z})\big(\bar{G}_{t,z} + \varepsilon_{t,z}\big)\Big] \\ &= \bar{G} + \sum_{t,z} \alpha_{t,z}(B_{t,z} + \Delta_{t,z}) + \sum_{t,z}(1-\alpha_{t,z})\varepsilon_{t,z} = W + \eta.\end{aligned}$$

Because W is $\mathcal{H}$-measurable and $\mathbb{E}[\eta \mid \mathcal{H}] = 0$, Equation (36) follows after subtracting $\mathbb{E}[\bar{G}] = \mathbb{E}[G_R^\gamma]$. Moreover $\mathbb{E}[(W - \mathbb{E}W)\eta^\top] = \mathbb{E}[(W - \mathbb{E}W)\mathbb{E}[\eta^\top \mid \mathcal{H}]] = 0$, so

$$\mathrm{Cov}(\widehat{G}_H) = \mathrm{Cov}(W) + \mathbb{E}[\mathrm{Cov}(\eta \mid \mathcal{H})].$$

The law of total covariance gives

$$\mathrm{Cov}(G_R^\gamma) = \mathrm{Cov}(\bar{G}) + \mathbb{E}[\mathrm{Cov}(\varepsilon \mid \mathcal{H})], \tag{39}$$

and subtraction yields Equation (38). The uniform-mixing statement follows from $\mathrm{Cov}(\eta \mid \mathcal{H}) = (1-\alpha)^2\,\mathrm{Cov}(\varepsilon \mid \mathcal{H})$. □

These results show that, even if the oracle causal and temporal estimators have the same mean, an input-dependent mixing need not preserve that mean: the weighted projection-gap and estimator-discrepancy terms in Equation (36) must still cancel in expectation.

Proposition E.4.2 (The multi-reward causal estimator is not generally a Rao–Blackwellization)**.** *Consider one initial binary action $A \in \{0,1\}$ with $\pi_\theta(A=1) = \mathrm{logistic}(\theta)$ evaluated at $\theta = 0$, and two future binary rewards R_1, R_2 that are conditionally independent given A, with*

$$\begin{aligned}&\mathbb{P}(R_1 = 1 \mid A = 0) = 0.1, \qquad &&\mathbb{P}(R_1 = 1 \mid A = 1) = 0.9, \\ &\mathbb{P}(R_2 = 1 \mid A = 0) = 0.8, \qquad &&\mathbb{P}(R_2 = 1 \mid A = 1) = 0.2.\end{aligned} \tag{40}$$

Let $U_k = R_k$, pool the two delays uniformly, and take $\gamma = 1$. Then

$$G_C^\star = 0.05(R_1 + R_2), \qquad G_R = \big(A - \tfrac{1}{2}\big)(R_1 + R_2), \qquad \mathbb{E}[G_C^\star] = \mathbb{E}[G_R] = 0.05, \tag{41}$$

yet $\mathbb{E}[G_R \mid G_C^\star = 0.05] = 0 \neq 0.05$. Consequently there is no sigma-field $\mathcal{H}$ with both $G_C^\star$ $\mathcal{H}$-measurable and $G_C^\star = \mathbb{E}[G_R \mid \mathcal{H}]$. The example satisfies Assumption E.3.1 with $m_0(u) = u$: unbiasedness does not imply a Rao–Blackwell identity.

Proof. At $\theta = 0$, $\pi(A = 1) = 1/2$, $\nabla_\theta \pi(1) = 1/4$ and $\nabla_\theta \pi(0) = -1/4$. For a uniformly selected future index K, $\mathbb{P}(U_K = 1 \mid A = 0) = 0.45$ and $\mathbb{P}(U_K = 1 \mid A = 1) = 0.55$, with marginal $1/2$, so Bayes' rule gives the pooled coefficients $w_0^\star(1, 1) = 0.1$ and $w_0^\star(0, 1) = -0.1$ (arguments (a, u); the single source state is suppressed). Terms with $U_k = 0$ are multiplied by $R_k = 0$, so each future reward contributes $\{\frac{1}{4}(0.1) - \frac{1}{4}(-0.1)\}R_k = 0.05R_k$, yielding the first identity in Equation (41); the second is the Bernoulli-logit score. Direct expectation gives $\mathbb{E}[G_C^\star] = 0.05(\mathbb{E}R_1 + \mathbb{E}R_2) = 0.05$ and $\mathbb{E}[G_R] = \frac{1}{4}(1.1) - \frac{1}{4}(0.9) = 0.05$.

Let $E := \{R_1 + R_2 = 1\}$. Conditional independence gives $\mathbb{P}(E \mid A = 0) = 0.1(0.2) + 0.9(0.8) = 0.74$ and $\mathbb{P}(E \mid A = 1) = 0.9(0.8) + 0.1(0.2) = 0.74$, so A remains Bernoulli$(1/2)$ conditional on E. Since $G_C^\star = 0.05$ exactly on E, $\mathbb{E}[G_R \mid G_C^\star = 0.05] = \mathbb{E}[A - \frac{1}{2} \mid E] = 0$. If some $\mathcal{H}$ satisfied the stated properties then $\sigma(G_C^\star) \subseteq \mathcal{H}$ and the tower property would give $\mathbb{E}[G_R \mid G_C^\star] = \mathbb{E}[\mathbb{E}[G_R \mid \mathcal{H}] \mid G_C^\star] = G_C^\star$, a contradiction. □

The proposition isolates the reason for the pair-local construction that follows. A single conditioning field for the summed estimator is too restrictive, whereas for each individual pair the scalar-reward encoding supplies the exact identity $\mathbb{E}[G_R^{(t,z)} \mid \mathcal{H}_{t,z}] = G_{CD}^{\star,(t,z)}$. Because the information set $\mathcal{H}_{t,z}$ changes with (t, z), those identities do not imply a global Rao–Blackwellization. Appendix E.5 therefore derives the mixing coefficient pair by pair and treats decoupling across pairs as an explicit approximation rather than as a consequence of conditioning.

E.5 Estimating the optimal mixing: pair-local derivation

The previous section gives an exact global decomposition for arbitrary data-dependent mixing coefficients, but it deliberately does not produce a closed-form HDCA mixing. We now derive that mixing coefficient from the pair-local object that is available under the scalar-reward encoding. Throughout this subsection, the ideal term is the delay-conditioned oracle $G_{CD}^{\star,(t,z)}$, matching the target of HDCA, rather than the pooled oracle $G_C^{\star,(t,z)}$ used as a global reference in Appendix E.4.

The derivation has three ingredients. First, under $U = R$ the delay-conditioned oracle term is exactly the conditional expectation of the corresponding REINFORCE pair under its own information set. Second, under explicit orthogonality and cross-pair decoupling assumptions, the hybrid MSE separates across pairs and yields a scalar shrinkage rule. Third, the two oracle quantities in that rule are replaced by aleatoric and epistemic quantities supplied by the hindsight ensemble.

Throughout, fix a pair $(t, z) \in \mathcal{I}_T$, abbreviate the hindsight input $x_{t,z} := (t, S_t, U_{t+z}, z)$, and write

$$g_a := \nabla_\theta \log \pi_\theta(a \mid S_t) \in \mathbb{R}^d, \qquad \Gamma_{ab} := \langle g_a, g_b \rangle, \qquad a, b \in \mathcal{A}, \tag{42}$$

for the per-action policy scores and their Gram matrix. Recall $\mathcal{V}(X) := \mathbb{E}\,\|X - \mathbb{E}X\|^2 = \operatorname{tr}\operatorname{Cov}(X)$ from Definition D.0.1.

Step 1: express the causal pair as a score average. The contribution coefficient converts the causal pair into a posterior average of policy scores, putting it on the same scale as the temporal pair.

Lemma E.5.1 (Score collapse)**.** *Let p be any probability distribution on $\mathcal{A}$ and let $w(a) := p(a)/\pi_\theta(a \mid S_t) - 1$. Then*

$$\sum_{a \in \mathcal{A}} \nabla_\theta \pi_\theta(a \mid S_t)\, w(a)\, R_{t+z} = R_{t+z} \sum_{a \in \mathcal{A}} g_a p(a). \tag{43}$$

Proof. Using $\nabla_\theta \pi_\theta = \pi_\theta \nabla_\theta \log \pi_\theta$, the term carrying p/π_θ equals $\sum_a g_a p(a)$. The term carrying -1 is $-\nabla_\theta \sum_a \pi_\theta(a \mid S_t) = -\nabla_\theta 1 = 0$. □

Applying Lemma E.5.1 with $p = h_{t,z}^\star$ and with $p = h_{\phi_m}$ gives

$$G_{CD}^{\star,(t,z)} = R_{t+z} \sum_a g_a h_{t,z}^\star(a \mid S_t, U_{t+z}), \qquad \widehat{G}_{CD}^{(t,z),m} = R_{t+z} \sum_a g_a h_{\phi_m}(a \mid x_{t,z}). \tag{44}$$

Step 2: identify the removable temporal variance. Let

$$\mathcal{H}_{t,z} := \sigma(S_t, U_{t+z}) \tag{45}$$

be the pair-local information set.

Lemma E.5.2 (Pair-local projection and residual variance)**.** *Under the scalar-reward encoding $U = R$ and full support (Equation* (16)*),*

$$\mathbb{E}\big[G_R^{(t,z)} \mid \mathcal{H}_{t,z}\big] = G_{CD}^{\star,(t,z)}, \tag{46}$$

and, writing $h := h_{t,z}^\star(\cdot \mid S_t, U_{t+z})$ as a vector on $\mathcal{A}$,

$$\vartheta_\varepsilon^{(t,z)} := \mathcal{V}\big(G_R^{(t,z)} \mid \mathcal{H}_{t,z}\big) = R_{t+z}^2 \operatorname{tr}\Big(\Gamma\big(\operatorname{diag}(h) - hh^\top\big)\Big). \tag{47}$$

If $\varepsilon_{t,z} := G_R^{(t,z)} - G_{CD}^{\star,(t,z)}$, then

$$\mathbb{E}\big[\vartheta_\varepsilon^{(t,z)}\big] = \mathbb{E}\,\|\varepsilon_{t,z}\|^2 =: V_\varepsilon^{(t,z)}. \tag{48}$$

Thus $V_\varepsilon^{(t,z)}$ is exactly the temporal variance that ideal pair-local causal conditioning removes.

Proof. Under $U = R$, both R_{t+z} and the scores g_a are $\mathcal{H}_{t,z}$-measurable. Conditional on $\mathcal{H}_{t,z}$, A_t has probabilities $h(a) := h_{t,z}^\star(a \mid S_t, U_{t+z})$. Writing $\mu_h := \sum_a h(a) g_a$, we obtain

$$\mathbb{E}[G_R^{(t,z)} \mid \mathcal{H}_{t,z}] = R_{t+z}\mu_h = G_{CD}^{\star,(t,z)}$$

by Equation (44). Thus $\varepsilon_{t,z} = R_{t+z}(g_{A_t} - \mu_h)$ and $\mathbb{E}[\varepsilon_{t,z} \mid \mathcal{H}_{t,z}] = 0$. In particular,

$$\begin{aligned}
\mathbb{E}[\|\varepsilon_{t,z}\|^2 \mid \mathcal{H}_{t,z}] &= R_{t+z}^2 \sum_a h(a)\,\|g_a - \mu_h\|^2 \\
&= R_{t+z}^2 \left(\sum_a h(a)\,\|g_a\|^2 - \|\mu_h\|^2\right) \\
&= R_{t+z}^2 \left(\sum_a h(a)\Gamma_{aa} - \sum_{a,b} h(a)h(b)\Gamma_{ab}\right) \\
&= R_{t+z}^2 \operatorname{tr}\big(\Gamma(\operatorname{diag}(h) - hh^\top)\big) = \vartheta_\varepsilon^{(t,z)}.
\end{aligned}$$

This is Equation (47). Taking an outer expectation now gives

$$\mathbb{E}[\vartheta_\varepsilon^{(t,z)}] = \mathbb{E}\Big[\mathbb{E}[\|\varepsilon_{t,z}\|^2 \mid \mathcal{H}_{t,z}]\Big] = \mathbb{E}\,\|\varepsilon_{t,z}\|^2 = V_\varepsilon^{(t,z)},$$

which proves Equation (48). Equivalently, total variance yields $\mathcal{V}(G_R^{(t,z)}) = \mathcal{V}(G_{CD}^{\star,(t,z)}) + V_\varepsilon^{(t,z)}$. □

Equation (46) does not contradict Proposition E.4.2. It is a family of identities, one for each pair and each pair's own sigma-field $\mathcal{H}_{t,z}$, and it uses the delay-conditioned coefficient $w_{t,z}^\star$. These pair-local conditional expectations cannot in general be summed into a conditional expectation with respect to one common sigma-field. Accordingly, Equation (47) is used to calibrate a local mixing, not to claim a covariance ordering for the full estimator; the exact global criterion remains Equation (38).

Step 3: derive the pairwise MSE-optimal shrinkage. We now state the two simplifying assumptions needed to make the joint MSE quadratic separable across pairs.

Assumption E.5.3 (Pairwise mixing conditions)**.** *Let*

$$\varepsilon_{t,z} := G_R^{(t,z)} - G_{CD}^{\star,(t,z)}, \qquad \Delta_{t,z} := \widehat{G}_{CD}^{(t,z)} - G_{CD}^{\star,(t,z)},$$

and write $G_{CD}^\star := \Pi + \sum_{(t,z)\in\mathcal{I}_T} G_{CD}^{\star,(t,z)}$. Set $V_\varepsilon^{(t,z)} := \mathbb{E}\,\|\varepsilon_{t,z}\|^2$ and $V_\Delta^{(t,z)} := \mathbb{E}\,\|\Delta_{t,z}\|^2$, assumed finite with $V_\varepsilon^{(t,z)} > 0$. For distinct pairs $(t,z) \neq (t',z')$, assume

(A1) ***Signal–error orthogonality.*** $\mathbb{E}\langle G_{CD}^\star - \mathbb{E}G_{CD}^\star, \Delta_{t,z}\rangle = 0$.

(A2) ***Cross-pair decoupling.*** $\mathbb{E}\langle G^\star_{CD} - \mathbb{E}G^\star_{CD}, \varepsilon_{t,z}\rangle = 0$ *and*

$$\mathbb{E}\langle \varepsilon_{t,z}, \varepsilon_{t',z'}\rangle = \mathbb{E}\langle \Delta_{t,z}, \Delta_{t',z'}\rangle = \mathbb{E}\langle \varepsilon_{t,z}, \Delta_{t',z'}\rangle = 0.$$

The within-pair cross terms need no additional assumption. Once the ensemble is trained, $G^{\star,(t,z)}_{CD}$ and $\Delta_{t,z}$ are $\mathcal{H}_{t,z}$-measurable, while $\mathbb{E}[\varepsilon_{t,z} \mid \mathcal{H}_{t,z}] = 0$ by Equation (46); hence $\mathbb{E}\langle G^{\star,(t,z)}_{CD}, \varepsilon_{t,z}\rangle = 0$ and $\mathbb{E}\langle \Delta_{t,z}, \varepsilon_{t,z}\rangle = 0$. Assumption (A2) is a simplification that permits coordinatewise optimization.

The general bias results in Appendix E.9 do not require either assumption.

Theorem E.5.4 (Pairwise MSE-optimal mixing coefficient)**.** *Under Assumption D.0.3,* $\gamma = 1$*, the pair-local projection Equation* (46)*, and Assumption E.5.3,* $\mathrm{MSE}(\alpha) := \mathbb{E}\left\|\widehat{G}_H - g\right\|^2$ *is a separable convex quadratic in the mixing vector. Its minimizer over deterministic mixing vectors* $\alpha \in [0,1]^{\mathcal{I}_T}$ *is coordinatewise*

$$\alpha^\star_{t,z} = \frac{V^{(t,z)}_\varepsilon}{V^{(t,z)}_\varepsilon + V^{(t,z)}_\Delta} = \left(1 + V^{(t,z)}_\Delta / V^{(t,z)}_\varepsilon\right)^{-1} \in (0,1]. \tag{49}$$

Proof. For brevity, index pairs by $i \in \mathcal{I}_T$, and put $C := G^\star_{CD} - g$ and $e_i := \alpha_i\Delta_i + (1-\alpha_i)\varepsilon_i$. Substitution into Equation (6) gives

$$\widehat{G}_H = \underbrace{\Pi + \sum_{i\in\mathcal{I}_T} G^{\star,i}_{CD}}_{G^\star_{CD}} + \sum_{i\in\mathcal{I}_T}\{\alpha_i\Delta_i + (1-\alpha_i)\varepsilon_i\}. \tag{50}$$

By Equation (46) and iterated expectation, $\mathbb{E}\varepsilon_i = 0$ and

$$\mathbb{E}[G^\star_{CD}] = \mathbb{E}[\Pi] + \sum_i \mathbb{E}\big[\mathbb{E}[G^i_R \mid \mathcal{H}_i]\big] = \mathbb{E}[G_R] = g,$$

where the last equality follows from Proposition E.1.1 at $\gamma = 1$. Thus $C = G^\star_{CD} - \mathbb{E}[G^\star_{CD}]$.

Fix any ordering of the finite pair set. Expanding the squared norm yields

$$\mathrm{MSE}(\alpha) = \mathbb{E}\|C\|^2 + 2\sum_i \mathbb{E}\langle C, e_i\rangle + \sum_i \mathbb{E}\|e_i\|^2 + 2\sum_{i<j}\mathbb{E}\langle e_i, e_j\rangle.$$

We check these terms separately. First, items (A1) and (A2) give

$$\mathbb{E}\langle C, e_i\rangle = \alpha_i\mathbb{E}\langle C, \Delta_i\rangle + (1-\alpha_i)\mathbb{E}\langle C, \varepsilon_i\rangle = 0.$$

Second, Δ_i is $\mathcal{H}_i$-measurable in the fixed-model experiment, whereas $\mathbb{E}[\varepsilon_i \mid \mathcal{H}_i] = 0$. Hence

$$\mathbb{E}\langle \Delta_i, \varepsilon_i\rangle = \mathbb{E}\langle \Delta_i, \mathbb{E}[\varepsilon_i \mid \mathcal{H}_i]\rangle = 0,$$

and therefore

$$\mathbb{E}\|e_i\|^2 = \alpha_i^2 V^i_\Delta + (1-\alpha_i)^2 V^i_\varepsilon.$$

Finally, for distinct i, j,

$$\begin{aligned}\mathbb{E}\langle e_i, e_j\rangle = {}& \alpha_i\alpha_j\mathbb{E}\langle \Delta_i, \Delta_j\rangle + (1-\alpha_i)(1-\alpha_j)\mathbb{E}\langle \varepsilon_i, \varepsilon_j\rangle \\ & + \alpha_i(1-\alpha_j)\mathbb{E}\langle \Delta_i, \varepsilon_j\rangle + (1-\alpha_i)\alpha_j\mathbb{E}\langle \varepsilon_i, \Delta_j\rangle = 0\end{aligned}$$

by item (A2), applied in both index orders for the mixed terms. Combining the preceding identities gives

$$\mathrm{MSE}(\alpha) = \mathbb{E}\|G^\star_{CD} - g\|^2 + \sum_{i\in\mathcal{I}_T}\left\{\alpha_i^2 V^i_\Delta + (1-\alpha_i)^2 V^i_\varepsilon\right\}. \tag{51}$$

For the ith summand $f_i(a) := a^2 V^i_\Delta + (1-a)^2 V^i_\varepsilon$,

$$f_i'(a) = 2a(V^i_\Delta + V^i_\varepsilon) - 2V^i_\varepsilon, \qquad f_i''(a) = 2(V^i_\Delta + V^i_\varepsilon) > 0.$$

Its unique stationary point is $a^\star_i = V^i_\varepsilon/(V^i_\varepsilon + V^i_\Delta)$. Since $V^i_\varepsilon > 0$ and $V^i_\Delta \geq 0$, this point lies in $(0,1]$ (including $a^\star_i = 1$ when $V^i_\Delta = 0$), so it is also the constrained minimizer on $[0,1]$. Separability proves Equation (49). □

Corollary E.5.5 (The local variance–error trade-off)**.** *Under the assumptions of Theorem E.5.4, for a fixed pair and any $\alpha \in (0, 1]$, the pair's contribution to MSE is lower than at the temporal endpoint $\alpha = 0$ if and only if*

$$\alpha V_{\Delta}^{(t,z)} < (2 - \alpha) V_{\varepsilon}^{(t,z)}. \tag{52}$$

If the learned error is centered, the same comparison is a variance comparison. Thus V_ε is the temporal variance available to remove, whereas V_Δ is the mean-squared error imported by trusting the learned causal estimator.

Proof. Subtract the $\alpha = 0$ term from the corresponding summand of Equation (51):

$$\alpha^2 V_{\Delta} + (1 - \alpha)^2 V_{\varepsilon} - V_{\varepsilon} = \alpha\{\alpha V_{\Delta} - (2 - \alpha) V_{\varepsilon}\}.$$

When $\mathbb{E}[\Delta_{t,z}] = 0$, the pairwise error is centered, so the MSE difference equals the variance difference. □

Step 4: estimate the epistemic and aleatoric terms. The learned-estimator error $\Delta_{t,z} = R_{t+z} \sum_a g_a(\bar{h}_\phi - h^\star_{t,z})(a)$ is unobservable. We estimate its scale by the spread of ensemble members around their mean:

$$\widetilde{V}_{\Delta}^{(t,z)} := \frac{1}{M-1} \sum_{m=1}^{M} \left\| R_{t+z} \sum_a g_a \big(h_{\phi_m} - \bar{h}_\phi\big)(a \mid x_{t,z}) \right\|^2 = R_{t+z}^2 \operatorname{tr}\big(\Gamma \widehat{\Sigma}_h(x_{t,z})\big), \tag{53}$$

where

$$\widehat{\Sigma}_h(x) := \frac{1}{M-1} \sum_m \big(h_{\phi_m}(x) - \bar{h}_\phi(x)\big)\big(h_{\phi_m}(x) - \bar{h}_\phi(x)\big)^\top.$$

The removable temporal variance in Equation (47) and the model-error proxy in Equation (53) are quadratic forms in the same policy-score Gram matrix Γ. Evaluating that matrix directly would require $|\mathcal{A}|$ policy backward passes per source state. Both quadratic forms, however, live on the zero-sum subspace $\mathbf{1}^\perp$: $(\operatorname{diag}(h) - hh^\top)\mathbf{1} = 0$, and each $h_{\phi_m} - \bar{h}_\phi$ sums to zero. We therefore use the local surrogate

$$\Gamma\big|_{\mathbf{1}^\perp} \approx cI, \qquad c > 0, \tag{54}$$

which is exact with $c = 1$ for a tabular softmax policy and approximate for a general parameterization. Indeed, for tabular softmax $g_a = e_a - \pi$, so for every $v \in \mathbf{1}^\perp$, $v^\top \Gamma v = \|\sum_a v_a(e_a - \pi)\|^2 = \|v\|^2$. Replacing the oracle posterior in Equation (47) by $\bar{h} := \bar{h}_\phi(x_{t,z})$ gives

$$\widetilde{V}_{\varepsilon}^{(t,z)} \approx cR_{t+z}^2\big(1 - \|\bar{h}\|_2^2\big), \qquad \widetilde{V}_{\Delta}^{(t,z)} \approx cR_{t+z}^2 \operatorname{tr} \widehat{\Sigma}_h(x_{t,z}). \tag{55}$$

We write $\widetilde{V}$ for these scaled surrogates; up to the common factor cR_{t+z}^2 they equal the scale-free surrogates $\widehat{V}_\varepsilon^{(t,z)}$ and $\widehat{V}_\Delta^{(t,z)}$ of Equation (8) (using $\operatorname{tr} \widehat{\Sigma}_h = \frac{M}{M-1}\mathcal{U}^2$, shown below).

Proposition E.5.6 (Practical uncertainty-adaptive mixing coefficient)**.** *Let $M \geq 2$. Substituting Equation* (55) *into the oracle shrinkage Equation* (49) *cancels the common policy-score and reward scales and gives*

$$\boxed{\widehat{\alpha}_{t,z} = \frac{1 - \|\bar{h}_\phi(x_{t,z})\|_2^2}{\big(1 - \|\bar{h}_\phi(x_{t,z})\|_2^2\big) + \frac{M}{M-1}\mathcal{U}^2(x_{t,z})} \in [0, 1].} \tag{56}$$

Here $\mathcal{U}^2$ is the ensemble disagreement; when numerator and denominator both vanish we set $\widehat{\alpha}_{t,z} := 1$. For fixed posterior predictions, the mixing coefficient is invariant to nonzero common rescalings of the explicit reward or policy-score factors in the variance surrogates.

Proof. Let $\bar{h} := \bar{h}_\phi(x_{t,z})$, $a := 1 - \|\bar{h}\|_2^2$, and $u := \mathcal{U}^2(x_{t,z})$. The sample-covariance trace satisfies

$$\operatorname{tr} \widehat{\Sigma}_h = \frac{1}{M-1} \sum_m \|h_{\phi_m} - \bar{h}\|_2^2 = \frac{M}{M-1} u.$$

When $cR_{t+z}^2 > 0$ and $a + \frac{M}{M-1}u > 0$, substitution into the variance-surrogate ratio gives

$$\frac{cR_{t+z}^2 a}{cR_{t+z}^2 a + cR_{t+z}^2 \frac{M}{M-1} u} = \frac{a}{a + \frac{M}{M-1}u},$$

which is Equation (56). If $R_{t+z} = 0$, both pair contributions are zero, so any mixing coefficient gives the same pair update; the scale-free formula specifies a convention at that input. If its own denominator is zero, the proposition prescribes $\widehat{\alpha}_{t,z} = 1$.

For probability-vector ensemble members,

$$u = \frac{1}{M}\sum_m \|h_{\phi_m}\|_2^2 - \left\|\bar{h}\right\|_2^2 \le 1 - \left\|\bar{h}\right\|_2^2 = a.$$

Thus for $a > 0$,

$$\frac{M-1}{2M-1} = \frac{1}{1 + M/(M-1)} \le \widehat{\alpha}_{t,z} \le 1.$$

When $a = 0$, the inequality $0 \le u \le a$ forces $u = 0$ and the chosen mixing coefficient is again one. Finally, with posterior predictions held fixed, a nonzero common reward or score rescaling multiplies both variance surrogates by the same positive squared factor and leaves the ratio unchanged. □

Interpretation and scope of the practical mixing coefficient. Equation (56) is literally an *aleatoric*/(*aleatoric*+*epistemic*) shrinkage. The term $1 - \left\|\bar{h}_\phi\right\|_2^2$ measures the residual source-action ambiguity that causal conditioning can remove; $\frac{M}{M-1}\mathcal{U}^2$ measures ensemble disagreement, used as a proxy for the error that trusting the learned estimator may import. Equivalently,

$$\widehat{\alpha}_{t,z} = \big(1 + \beta_{\text{eff}}(x_{t,z})\mathcal{U}^2(x_{t,z})\big)^{-1}, \qquad \beta_{\text{eff}}(x) := \frac{M}{(M-1)\big(1 - \left\|\bar{h}_\phi(x)\right\|_2^2\big)}. \tag{57}$$

When $1 - \left\|\bar{h}_\phi\right\|_2^2 > 0$, the practical mixing depends on the ratio of ensemble disagreement to the aleatoric surrogate, rather than on disagreement alone. For the same positive disagreement, a smaller aleatoric surrogate yields a smaller mixing coefficient. However, $\mathcal{U}^2 \le 1 - \left\|\bar{h}_\phi\right\|_2^2$ implies $\widehat{\alpha}_{t,z} \ge (M-1)/(2M-1)$, so this practical rule cannot close all the way to zero. At an exact point-mass ensemble mean, both quantities vanish and the chosen convention sets the mixing coefficient to one.

Relative to the oracle rule Equation (49), the practical mixing coefficient makes four explicit approximations. (i) It uses the isotropic score surrogate $\Gamma|_{\mathbf{1}^\perp} \approx cI$. (ii) It replaces the oracle posterior in the aleatoric term by the ensemble mean. (iii) It uses ensemble spread as a proxy for V_Δ. (iv) It localizes population second moments to the current pair input and applies the coordinatewise oracle rule under the separability assumptions of Assumption E.5.3.

These approximations do not affect the general bias analysis bound of Proposition E.9.1, which holds for any mixing vector. The asymptotic results in Appendix E.9 additionally require the mixing-coefficient contraction condition Equation (74); the practical mixing satisfies it under the following mild non-degeneracy condition.

Corollary E.5.7 (Contraction of the practical mixing coefficient)**.** *Suppose the aleatoric term is uniformly bounded away from zero on the evaluated inputs: for some $c_0 > 0$ independent of iteration n,*

$$1 - \left\|\bar{h}_\phi(x_{t,z})\right\|_2^2 \ge c_0.$$

Define $\beta_{\max} := M/\{(M-1)c_0\}$. Then

$$0 \le 1 - \widehat{\alpha}_{t,z} = \frac{\beta_{\text{eff}}\mathcal{U}^2}{1 + \beta_{\text{eff}}\mathcal{U}^2} \le \frac{\beta_{\max}\mathcal{U}^2}{1 + \beta_{\max}\mathcal{U}^2} \le \sqrt{\beta_{\max}}\,\mathcal{U}. \tag{58}$$

At iteration n, let $\mathcal{U}_{t,z,n} \ge 0$ denote the disagreement used at pair (t,z), and define

$$\bar{\mathcal{U}}_n := \max_{(t,z)\in\mathcal{I}_T} \{\mathbb{E}[\mathcal{U}_{t,z,n}^2 \mid \mathcal{F}_n]\}^{1/2}.$$

Then,

$$\|1 - \widehat{\alpha}_n\|_{\bullet,\mathcal{F}_n} \le \sqrt{\beta_{\max}}\,\bar{\mathcal{U}}_n.$$

Hence Equation (74) *holds with $C_\alpha = \sqrt{\beta_{\max}}$. If the remaining assumptions of Theorem E.9.2 hold, that theorem applies; with the additional rate assumptions, Corollary E.9.3 applies as well.*

Proof. Write $a := 1 - \left\|\bar{h}_\phi(x_{t,z})\right\|_2^2 \geq c_0$ and $k := M/(M-1)$. Then $\beta_{\text{eff}} = k/a \leq k/c_0 = \beta_{\max}$. By Equation (56),

$$0 \leq 1 - \widehat{\alpha}_{t,z} = \frac{\beta_{\text{eff}}\mathcal{U}^2}{1 + \beta_{\text{eff}}\mathcal{U}^2} \leq \frac{\beta_{\max}\mathcal{U}^2}{1 + \beta_{\max}\mathcal{U}^2}.$$

For $y \geq 0$, $y/(1+y) \leq \min\{y, 1\} \leq \sqrt{y}$. Taking $y = \beta_{\max}\mathcal{U}^2$ proves $1 - \widehat{\alpha}_{t,z} \leq \sqrt{\beta_{\max}}\mathcal{U}$. At iteration n, square this pointwise inequality and condition on $\mathcal{F}_n$:

$$\mathbb{E}[(1 - \widehat{\alpha}_{t,z,n})^2 \mid \mathcal{F}_n] \leq \beta_{\max}\mathbb{E}[\mathcal{U}_{t,z,n}^2 \mid \mathcal{F}_n].$$

Taking square roots and then the maximum over the finite pair set gives

$$\|1 - \widehat{\alpha}_n\|_{\bullet,\mathcal{F}_n} \leq \sqrt{\beta_{\max}} \max_{(t,z)\in\mathcal{I}_T} \{\mathbb{E}[\mathcal{U}_{t,z,n}^2 \mid \mathcal{F}_n]\}^{1/2} = \sqrt{\beta_{\max}}\bar{\mathcal{U}}_n.$$

This is Equation (74) with $C_\alpha = \sqrt{\beta_{\max}}$. The asymptotic and rate conclusions additionally require the other assumptions of their respective results. □

E.6 Bias of adaptive mixing

Theorem E.6.1 (Exact decomposition of adaptive hybrid bias)**.** *Assume g is $\mathcal{F}$-measurable. Define $b_R := \mathbb{E}[G_R^\gamma - g \mid \mathcal{F}]$ and, for each pair, the conditional estimator-mean discrepancy and its centered fluctuation*

$$d_{t,z} := \mathbb{E}\big[\widehat{G}_{CD}^{(t,z)} - G_R^{\gamma,(t,z)} \,\big|\, \mathcal{F}\big], \qquad \zeta_{t,z} := \big(\widehat{G}_{CD}^{(t,z)} - G_R^{\gamma,(t,z)}\big) - d_{t,z}. \tag{59}$$

Then for any mixing vector $\alpha \in [0,1]^{\mathcal{I}_T}$,

$$b_H := \mathbb{E}[\widehat{G}_H - g \mid \mathcal{F}] = b_R + \sum_{(t,z)\in\mathcal{I}_T} \Big\{\mathbb{E}[\alpha_{t,z} \mid \mathcal{F}]\, d_{t,z} + \mathrm{Cov}_{\mathcal{F}}\big(\alpha_{t,z}, \zeta_{t,z}\big)\Big\}, \tag{60}$$

where $\mathrm{Cov}_{\mathcal{F}}(A, X) := \mathbb{E}[(A - \mathbb{E}[A \mid \mathcal{F}])(X - \mathbb{E}[X \mid \mathcal{F}]) \mid \mathcal{F}]$ for scalar A and vector X. For a uniform mixing and $\gamma = 1$ this reduces to $b_H = \mathbb{E}[\alpha \mid \mathcal{F}]b_{CD} + \mathrm{Cov}_{\mathcal{F}}(\alpha, \xi_{CD} - \xi_R)$ with $b_{CD} := \mathbb{E}[\widehat{G}_{CD} - g \mid \mathcal{F}]$ and ξ_{CD}, ξ_R the conditionally centered estimator noises.

Proof. From Equation (6),

$$\widehat{G}_H - g = \big(G_R^\gamma - g\big) + \sum_{t,z} \alpha_{t,z}\big(\widehat{G}_{CD}^{(t,z)} - G_R^{\gamma,(t,z)}\big), \tag{61}$$

because the shared term Π appears in both estimators and cancels from every pairwise difference. Taking $\mathbb{E}[\cdot \mid \mathcal{F}]$ gives $b_H = b_R + \sum_{t,z}\mathbb{E}[\alpha_{t,z}(d_{t,z} + \zeta_{t,z}) \mid \mathcal{F}]$. Since $d_{t,z}$ is $\mathcal{F}$-measurable and $\mathbb{E}[\zeta_{t,z} \mid \mathcal{F}] = 0$,

$$\mathbb{E}[\alpha_{t,z}(d_{t,z} + \zeta_{t,z}) \mid \mathcal{F}] = \mathbb{E}[\alpha_{t,z} \mid \mathcal{F}]d_{t,z} + \mathbb{E}\big[(\alpha_{t,z} - \mathbb{E}[\alpha_{t,z} \mid \mathcal{F}])\zeta_{t,z} \,\big|\, \mathcal{F}\big], \tag{62}$$

and the second term is $\mathrm{Cov}_{\mathcal{F}}(\alpha_{t,z}, \zeta_{t,z})$ by conditional centring of $\zeta_{t,z}$. This is Equation (60). For a uniform mixing, $\sum_{t,z} d_{t,z} = b_{CD} - b_R$ and $\sum_{t,z}\zeta_{t,z} = \xi_{CD} - \xi_R$, and at $\gamma = 1$, $b_R = 0$. □

E.7 Propagation of hindsight-posterior error

In this section, we analyze how hindsight-posterior estimation error propagates to the pooled causal gradient estimator. The oracle and learned estimators use the same trajectories, rewards, immediate-reward term, and policy scores; they differ only in whether $h_t^\star$ or $\widehat{h}_t$ is used in the future-reward coefficients. The same argument extends to delay conditioning, relating the posterior error $\widehat{h}_{t,z} - h_{t,z}^\star$ to the corresponding error $\widehat{G}_{CD}^{(t,z)} - G_{CD}^{\star,(t,z)}$ in each causal gradient contribution.

Theorem E.7.1 (Posterior error propagates linearly to causal-gradient bias)**.** *Let $\mathcal{F}$ be any sigma-field with respect to which the current policy and learned hindsight-model parameters are measurable. Under Assumption D.0.4,*

$$\Big\|\mathbb{E}[\widehat{G}_C - G_C^\star \mid \mathcal{F}]\Big\| \leq G_{\max}R_{\max}\,\mathbb{E}\Big[\sum_{0\leq t<k\leq T-1}\Big\|\widehat{h}_t(\cdot \mid S_t, U_k) - h_t^\star(\cdot \mid S_t, U_k)\Big\|_1 \,\Big|\, \mathcal{F}\Big]. \tag{63}$$

Defining the conditional average on-policy posterior error over the $N_T = T(T-1)/2$ future pairs,

$$\varepsilon_h(\mathcal{F}) := \frac{1}{N_T}\mathbb{E}\Big[\sum_{0\le t<k\le T-1}\Big\|\widehat{h}_t(\cdot \mid S_t, U_k) - h_t^\star(\cdot \mid S_t, U_k)\Big\|_1 \;\Big|\; \mathcal{F}\Big], \tag{64}$$

we obtain $\Big\|\mathbb{E}[\widehat{G}_C - G_C^\star \mid \mathcal{F}]\Big\| \le K_T\varepsilon_h(\mathcal{F})$ with $K_T := \frac{T(T-1)}{2}G_{\max}R_{\max}$. No factor $1/\pi_{\min}$ appears because the probability ratio in the coefficient cancels against $\nabla_\theta\pi = \pi\nabla_\theta\log\pi$.

Proof. The shared immediate-reward term cancels. Write $e_{t,k}(a) := \widehat{h}_t(a \mid S_t, U_k) - h_t^\star(a \mid S_t, U_k)$. Using $\nabla_\theta\pi_\theta/\pi_\theta = \nabla_\theta\log\pi_\theta$ gives

$$\widehat{G}_C - G_C^\star = \sum_{t<k} R_k \sum_{a\in\mathcal{A}} \nabla_\theta \log\pi_\theta(a \mid S_t)e_{t,k}(a). \tag{65}$$

Conditional Jensen, the triangle inequality, and Assumption D.0.4 now give, in that order,

$$\begin{aligned}
\Big\|\mathbb{E}[\widehat{G}_C - G_C^\star \mid \mathcal{F}]\Big\| &\le \mathbb{E}[\Big\|\widehat{G}_C - G_C^\star\Big\| \mid \mathcal{F}] \\
&\le \mathbb{E}\left[\sum_{t<k}|R_k|\sum_a \|\nabla_\theta\log\pi_\theta(a \mid S_t)\|\,|e_{t,k}(a)| \;\middle|\; \mathcal{F}\right] \\
&\le G_{\max}R_{\max}\,\mathbb{E}\left[\sum_{t<k}\|e_{t,k}\|_1 \;\middle|\; \mathcal{F}\right] \\
&= G_{\max}R_{\max}N_T\varepsilon_h(\mathcal{F}) = K_T\varepsilon_h(\mathcal{F}).
\end{aligned}$$

The third line is Equation (63); the last line uses Equation (64) and $N_T = T(T-1)/2$. The cancellation in Equation (65) explains why no inverse minimum action probability enters the bound. □

Remark E.7.2 (The same bound holds for the delay-conditioned estimator). Theorem E.7.1 is written for the pooled estimator $(\widehat{G}_C, G_C^\star)$ and the posterior pair $(\widehat{h}_t, h_t^\star)$, but the proof uses only two facts: that the two estimators differ solely in the posterior used in the future-reward coefficients, and that $\nabla_\theta\pi/\pi = \nabla_\theta\log\pi$ cancels the probability ratio. Both hold verbatim for the delay-conditioned estimator. Replacing $(\widehat{G}_C, G_C^\star)$ by $(\widehat{G}_{CD}, G_{CD}^\star)$, indexing future pairs by $(t,z) \in \mathcal{I}_T$, and replacing $(\widehat{h}_t, h_t^\star)$ by $(\widehat{h}_{t,z}, h_{t,z}^\star)$ therefore gives the same bound. Accordingly, Equations (63) and (64) and Corollary E.7.3 apply to either estimator, with ε_h interpreted as the average ℓ_1 error of the posterior family used by that estimator; the number of summands is $N_T = T(T-1)/2$ in both cases, so K_T is unchanged.

Corollary E.7.3 (Consistency transfer from hindsight prediction to causal gradient)**.** *Adopt Assumptions D.0.3 and D.0.4. At iteration n, condition on $\mathcal{F}_n$ and evaluate on a fresh current-policy batch. Let $b_{C,n}^{\mathrm{str}} := \mathbb{E}[G_{C,n}^\star - g_n \mid \mathcal{F}_n]$. Then the learned causal bias $b_{C,n} := \mathbb{E}[\widehat{G}_{C,n} - g_n \mid \mathcal{F}_n]$ satisfies*

$$\|b_{C,n}\| \le \big\|b_{C,n}^{\mathrm{str}}\big\| + K_T\varepsilon_{h,n}, \qquad \varepsilon_{h,n} := \varepsilon_h(\mathcal{F}_n). \tag{66}$$

Under Assumption E.3.1 (or Assumption F.1.1 for the delay-conditioned estimator) we have $b_{C,n}^{\mathrm{str}} = 0$, so $\varepsilon_{h,n} \to 0$ implies $b_{C,n} \to 0$. By Remark E.7.2, the identical statement holds for the delay-conditioned estimator with

$$b_{CD,n}^{\mathrm{str}} := \mathbb{E}[G_{CD,n}^\star - g_n \mid \mathcal{F}_n], \qquad b_{CD,n} := \mathbb{E}[\widehat{G}_{CD,n} - g_n \mid \mathcal{F}_n].$$

Under Assumption F.1.1, $b_{CD,n}^{\mathrm{str}} = 0$.

Proof. Add and subtract the pooled oracle estimator inside the conditional expectation:

$$b_{C,n} = \mathbb{E}[\widehat{G}_{C,n} - G_{C,n}^\star \mid \mathcal{F}_n] + \mathbb{E}[G_{C,n}^\star - g_n \mid \mathcal{F}_n] = \mathbb{E}[\widehat{G}_{C,n} - G_{C,n}^\star \mid \mathcal{F}_n] + b_{C,n}^{\mathrm{str}}.$$

Therefore, by the triangle inequality and Theorem E.7.1,

$$\begin{aligned}
\|b_{C,n}\| &\le \Big\|\mathbb{E}[\widehat{G}_{C,n} - G_{C,n}^\star \mid \mathcal{F}_n]\Big\| + \big\|b_{C,n}^{\mathrm{str}}\big\| \\
&\le K_T\varepsilon_{h,n} + \big\|b_{C,n}^{\mathrm{str}}\big\|.
\end{aligned}$$

Conditional on $\mathcal{F}_n$, the current policy is fixed and the evaluation batch is on-policy. Under Assumption E.3.1, Theorem E.3.3 consequently gives $\mathbb{E}[G^\star_{C,n} \mid \mathcal{F}_n] = g_n$, so $\|b_{C,n}\| \le K_T \varepsilon_{h,n} \to 0$ whenever $\varepsilon_{h,n} \to 0$.

For the delay-conditioned estimator the same decomposition is

$$b_{CD,n} = \mathbb{E}[\widehat{G}_{CD,n} - G^\star_{CD,n} \mid \mathcal{F}_n] + b^{\mathrm{str}}_{CD,n}.$$

Using Remark E.7.2 gives

$$\|b_{CD,n}\| \le K_T \varepsilon_{h,n} + \left\|b^{\mathrm{str}}_{CD,n}\right\|,$$

where $\varepsilon_{h,n}$ now uses delay-conditioned posteriors. Under Assumption F.1.1, Theorem F.1.3 gives $b^{\mathrm{str}}_{CD,n} = 0$, proving the stated consistency transfer. □

Theorem E.7.1 supplies the bridge between supervised hindsight learning and policy optimization: posterior consistency is not merely a predictive statement, since under bounded rewards and scores its error is amplified by at most the explicit constant K_T when transferred to causal-gradient bias. It is also the reason the detectability result of Theorem F.4.1 has policy-gradient consequences: a coefficient that a learner cannot resolve contributes posterior error, which Equation (63) converts linearly into gradient bias.

E.8 Hindsight consistency and vanishing adaptive bias

Draw a source–delay pair $(\mathsf{T}_n, \mathsf{Z}_n)$ uniformly from $\mathcal{I}_T$ (equivalently, draw the source time with probability H_t/N_T and then the delay uniformly), independently of the current trajectory given $\mathcal{F}_n$, and set $\mathsf{K}_n := \mathsf{T}_n + \mathsf{Z}_n$. For the delay-conditioned estimator define $X_n^{\mathrm{CD}} := (\mathsf{T}_n, S_{\mathsf{T}_n}, U_{\mathsf{K}_n}, \mathsf{Z}_n)$, whereas for the pooled estimator define $X_n^{\mathrm{pool}} := (\mathsf{T}_n, S_{\mathsf{T}_n}, U_{\mathsf{K}_n})$. In what follows, X_n denotes the input appropriate to the estimator under analysis, with conditional law μ_n, and $h^\star_n$ denotes its corresponding population posterior. For HDCA, $X_n = X_n^{\mathrm{CD}}$. Let $\bar{h}_n := M^{-1}\sum_m h_{m,n}$; throughout this subsection $\widehat{h}_n = \bar{h}_n$. Then

$$\varepsilon_{h,n} := \left\|\bar{h}_n - h^\star_n\right\|_{L^1(\mu_n)} = \mathbb{E}\left[\left\|\bar{h}_n(X_n) - h^\star_n(X_n)\right\|_1 \mid \mathcal{F}_n\right], \tag{67}$$

which agrees exactly with $\varepsilon_h(\mathcal{F}_n)$ in Equation (64). Convergence of $\mathcal{F}_n$-measurable quantities is understood almost surely with respect to the training history; the same implications hold in probability throughout.

Definition E.8.1 (Hindsight consistency and uncertainty reliability)**.** *Hindsight consistency* means $\varepsilon_{h,n} \to 0$ under the current on-policy evaluation distribution. Define

$$\bar{\mathcal{U}}_n := \max_{(t,z)\in\mathcal{I}_T} \left\{\mathbb{E}\left[\mathcal{U}^2_{t,z,n} \mid \mathcal{F}_n\right]\right\}^{1/2},$$

where $\mathcal{U}_{t,z,n}$ is the ensemble disagreement at source–delay pair (t,z) at iteration n. *Uncertainty contraction* means $\bar{\mathcal{U}}_n \to 0$. Consistency of the individual ensemble members implies both hindsight consistency and uncertainty contraction, as shown below.

Assumption E.8.2 (Hindsight posterior consistency under policy drift)**.** *For each n, let h_n^{cur} be a population-risk minimizer within the model class under μ_n, h_n^{train} the population-risk minimizer under the distribution actually used to train the auxiliary model, and $\widetilde{h}_n$ an exact minimizer of the empirical training objective. Assume the approximation, tracking, statistical and optimization errors*

$$a_n := \|h_n^{\mathrm{cur}} - h_n^\star\|_{L^1(\mu_n)}, \qquad d_n := \left\|h_n^{\mathrm{train}} - h_n^{\mathrm{cur}}\right\|_{L^1(\mu_n)}, \tag{68}$$

$$s_n := \left\|\widetilde{h}_n - h_n^{\mathrm{train}}\right\|_{L^1(\mu_n)}, \qquad o_n := \left\|\bar{h}_n - \widetilde{h}_n\right\|_{L^1(\mu_n)} \tag{69}$$

all converge to zero. By the triangle inequality $\varepsilon_{h,n} \le o_n + s_n + d_n + a_n \to 0$.

For the kernel-smoothed implementation, the approximation term a_n also includes the deviation of the kernel target from the delay-conditioned posterior. Thus, when the asymptotic consistency result is invoked for the smoothed method, its assumptions require this smoothing component to vanish asymptotically. Consider tabular maximum-likelihood estimation with a correctly specified categorical model on a finite input space, conditionally independent samples, and exact empirical optimization. A sufficient coverage condition for $s_n \to 0$ is then that the number of observations of every

tuple whose on-policy probability is bounded away from zero diverges. A sufficient controlled-drift condition is $\left\|h^\star_{n+1} - h^\star_n\right\|_{L^1(\mu_n)} \le L_h \left\|\theta_{n+1} - \theta_n\right\| \to 0$ together with an auxiliary-learning timescale for which the population target of the auxiliary data tracks the current target, yielding $d_n \to 0$.

Lemma E.8.3 (Consistent ensemble members imply uncertainty contraction)**.** *With $\bar{h}_n$ and $\mathcal{U}^2_n$ as in Proposition E.5.6, for every target $h^\star_n(x)$,*

$$\frac{1}{M}\sum_{m=1}^{M} \left\|h_{m,n}(x) - h^\star_n(x)\right\|^2 = \mathcal{U}^2_n(x) + \left\|\bar{h}_n(x) - h^\star_n(x)\right\|^2. \tag{70}$$

Hence if the average member error vanishes, $M^{-1}\sum_m \mathbb{E}[\|h_{m,n}(X_n) - h^\star_n(X_n)\|^2 \mid \mathcal{F}_n] \to 0$, then both $\mathbb{E}[\mathcal{U}^2_n(X_n) \mid \mathcal{F}_n] \to 0$ and $\mathbb{E}[\left\|\bar{h}_n(X_n) - h^\star_n(X_n)\right\|^2 \mid \mathcal{F}_n] \to 0$. Because $\mathcal{A}$ is finite,

$$\varepsilon_{h,n} \le \sqrt{|\mathcal{A}|}\left\{\mathbb{E}[\left\|\bar{h}_n(X_n) - h^\star_n(X_n)\right\|^2 \mid \mathcal{F}_n]\right\}^{1/2} \longrightarrow 0. \tag{71}$$

Proof. Fix x and abbreviate $d_m := h_{m,n}(x) - \bar{h}_n(x)$ and $e := \bar{h}_n(x) - h^\star_n(x)$. Since $M^{-1}\sum_m d_m = 0$,

$$\begin{aligned}
\frac{1}{M}\sum_m \|h_{m,n}(x) - h^\star_n(x)\|_2^2 &= \frac{1}{M}\sum_m \|d_m + e\|_2^2 \\
&= \frac{1}{M}\sum_m \|d_m\|_2^2 + 2\left\langle \frac{1}{M}\sum_m d_m, e\right\rangle + \|e\|_2^2 \\
&= \mathcal{U}^2_n(x) + \left\|\bar{h}_n(x) - h^\star_n(x)\right\|_2^2.
\end{aligned}$$

This proves Equation (70). Define the conditional average member error

$$D_n := \frac{1}{M}\sum_m \mathbb{E}[\|h_{m,n}(X_n) - h^\star_n(X_n)\|_2^2 \mid \mathcal{F}_n].$$

Taking conditional expectations in the pointwise identity yields

$$D_n = \mathbb{E}[\mathcal{U}^2_n(X_n) \mid \mathcal{F}_n] + \mathbb{E}[\left\|\bar{h}_n(X_n) - h^\star_n(X_n)\right\|_2^2 \mid \mathcal{F}_n].$$

Both summands are nonnegative, so each is at most D_n and tends to zero when $D_n \to 0$. Writing $v_n := \bar{h}_n(X_n) - h^\star_n(X_n)$, Cauchy–Schwarz first over actions and then over the conditional input law gives

$$\begin{aligned}
\varepsilon_{h,n} &= \mathbb{E}\left[\sum_{a\in\mathcal{A}} |v_n(a)| \,\middle|\, \mathcal{F}_n\right] \\
&\le \sqrt{|\mathcal{A}|}\,\mathbb{E}[\|v_n\|_2 \mid \mathcal{F}_n] \\
&\le \sqrt{|\mathcal{A}|}\,\{\mathbb{E}[\|v_n\|_2^2 \mid \mathcal{F}_n]\}^{1/2} \le \sqrt{|\mathcal{A}| D_n} \longrightarrow 0.
\end{aligned}$$

To relate the average disagreement to the quantity used by the mixing coefficient, let $u_{t,z,n} := \mathbb{E}[\mathcal{U}^2_{t,z,n} \mid \mathcal{F}_n]$. Because X_n is generated by a uniform draw from the N_T pairs,

$$\mathbb{E}[\mathcal{U}^2_n(X_n) \mid \mathcal{F}_n] = \frac{1}{N_T}\sum_{(t,z)\in\mathcal{I}_T} u_{t,z,n}.$$

For the fixed finite horizon,

$$\bar{\mathcal{U}}^2_n = \max_{(t,z)\in\mathcal{I}_T} u_{t,z,n} \le \sum_{(t,z)\in\mathcal{I}_T} u_{t,z,n} = N_T\mathbb{E}[\mathcal{U}^2_n(X_n) \mid \mathcal{F}_n] \le N_T D_n \longrightarrow 0.$$

Thus member consistency also implies the maximum-over-pairs uncertainty contraction required below. □

E.9 BIAS BOUNDS RELATIVE TO THE TEMPORAL AND CAUSAL ESTIMATORS

Proposition E.9.1 (Two-sided control of hybrid bias)**.** *Suppose Assumptions D.0.3 and D.0.5 hold, and evaluate on a fresh current-policy batch conditional on $\mathcal{F}_n$. Let $b_{H,n} := \mathbb{E}[\widehat{G}_{H,n} - g_n \mid \mathcal{F}_n]$ and $b_{CD,n} := \mathbb{E}[\widehat{G}_{CD,n} - g_n \mid \mathcal{F}_n]$. Then*

$$\|b_{H,n}\| \le \|b_R(\gamma)\| + C_G \|\alpha_n\|_{\bullet,\mathcal{F}_n}, \tag{72}$$

$$\|b_{H,n}\| \le \|b_{CD,n}\| + C_G \|1-\alpha_n\|_{\bullet,\mathcal{F}_n}, \tag{73}$$

where $b_R(\gamma)$ is evaluated at the current policy θ_n. Thus the smaller of the two right-hand sides is also a bound. Under Assumption D.0.4, Corollary E.7.3 and Remark E.7.2 further give

$$\|b_{H,n}\| \le \left\|b^{\mathrm{str}}_{CD,n}\right\| + K_T \varepsilon_{h,n} + C_G \|1-\alpha_n\|_{\bullet,\mathcal{F}_n}.$$

When $b^{\mathrm{str}}_{CD,n} = 0$, the structural term drops out. Neither bound requires independence between the mixing coefficient and the current estimator discrepancy.

Proof. Write $D_{t,z,n} := \widehat{G}^{(t,z)}_{CD,n} - G^{\gamma,(t,z)}_{R,n}$. For any nonnegative scalar weight $v_{t,z}$, conditional Jensen and Cauchy–Schwarz give

$$\begin{aligned}\|\mathbb{E}[v_{t,z} D_{t,z,n} \mid \mathcal{F}_n]\| &\le \mathbb{E}[v_{t,z} \|D_{t,z,n}\| \mid \mathcal{F}_n] \\ &\le \{\mathbb{E}[v^2_{t,z} \mid \mathcal{F}_n]\}^{1/2} \{\mathbb{E}[\|D_{t,z,n}\|^2 \mid \mathcal{F}_n]\}^{1/2} \\ &\le C_{t,z} \{\mathbb{E}[v^2_{t,z} \mid \mathcal{F}_n]\}^{1/2},\end{aligned}$$

where the last line uses Assumption D.0.5. This calculation does not factor an expectation of a product.

The temporal-endpoint decomposition is

$$\widehat{G}_{H,n} = G^{\gamma}_{R,n} + \sum_{t,z} \alpha_{t,z,n} D_{t,z,n}.$$

Conditional on $\mathcal{F}_n$, Proposition E.1.1 gives $\mathbb{E}[G^{\gamma}_{R,n} - g_n \mid \mathcal{F}_n] = b_R(\gamma)$ at θ_n. Consequently,

$$\begin{aligned}\|b_{H,n}\| &\le \|b_R(\gamma)\| + \sum_{t,z} \|\mathbb{E}[\alpha_{t,z,n} D_{t,z,n} \mid \mathcal{F}_n]\| \\ &\le \|b_R(\gamma)\| + \sum_{t,z} C_{t,z} \{\mathbb{E}[\alpha^2_{t,z,n} \mid \mathcal{F}_n]\}^{1/2} \\ &\le \|b_R(\gamma)\| + C_G \|\alpha_n\|_{\bullet,\mathcal{F}_n}.\end{aligned}$$

For the causal endpoint, the exact identity is

$$\widehat{G}_{H,n} = \widehat{G}_{CD,n} - \sum_{t,z} (1-\alpha_{t,z,n}) D_{t,z,n}.$$

Subtracting g_n, taking conditional expectations, and using the same product bound with $v_{t,z} = 1-\alpha_{t,z,n}$ yields

$$\begin{aligned}\|b_{H,n}\| &\le \|b_{CD,n}\| + \sum_{t,z} \|\mathbb{E}[(1-\alpha_{t,z,n}) D_{t,z,n} \mid \mathcal{F}_n]\| \\ &\le \|b_{CD,n}\| + \sum_{t,z} C_{t,z} \{\mathbb{E}[(1-\alpha_{t,z,n})^2 \mid \mathcal{F}_n]\}^{1/2} \\ &\le \|b_{CD,n}\| + C_G \|1-\alpha_n\|_{\bullet,\mathcal{F}_n}.\end{aligned}$$

These are Equations (72) and (73). Substituting $\|b_{CD,n}\| \le \left\|b^{\mathrm{str}}_{CD,n}\right\| + K_T \varepsilon_{h,n}$ from Corollary E.7.3 and Remark E.7.2 proves the final bound. □

These bounds hold for arbitrary convex mixing vectors. For the practical mixing coefficient in Equation (8), the causal weight is bounded away from zero, so the temporal-reference bound does not establish convergence to REINFORCE or arbitrarily small bias through mixing closure.

Theorem E.9.2 (Asymptotic unbiasedness under mixing-coefficient contraction)**.** *Suppose Assumptions D.0.3 to D.0.5 hold, and suppose the causal estimator has zero structural bias at every iteration: $b^{\mathrm{str}}_{CD,n} = 0$ for all n. For HDCA this follows from Assumption F.1.1, in particular when $U = R$. The pooled analogue uses Assumption E.3.1. Suppose further that the mixing coefficient satisfies, for some constant $C_\alpha < \infty$ independent of n,*

$$\|1 - \alpha_n\|_{\bullet,\mathcal{F}_n} \le C_\alpha \bar{\mathcal{U}}_n. \tag{74}$$

Then

$$\|b_{H,n}\| \le K_T \varepsilon_{h,n} + C_\alpha C_G \bar{\mathcal{U}}_n. \tag{75}$$

Consequently, if $\varepsilon_{h,n} \to 0$ and $\bar{\mathcal{U}}_n \to 0$ (as we analyzed in Appendix E.8), then $b_{H,n} \to 0$. In particular, learned-posterior error, dependence between the mixing coefficient and estimator noise, and discounting bias of the temporal estimator do not induce a nonzero asymptotic bias floor.

Proof. The late-regime bound Equation (73), posterior-consistency transfer, and mixing-coefficient contraction give, respectively,

$$\begin{aligned}\|b_{H,n}\| &\le \|b_{CD,n}\| + C_G \|1 - \alpha_n\|_{\bullet,\mathcal{F}_n} \\ &\le K_T \varepsilon_{h,n} + C_G \|1 - \alpha_n\|_{\bullet,\mathcal{F}_n} \\ &\le K_T \varepsilon_{h,n} + C_\alpha C_G \bar{\mathcal{U}}_n.\end{aligned}$$

The second line uses $b^{\mathrm{str}}_{CD,n} = 0$; the third uses Equation (74). This proves Equation (75). With fixed T and constants independent of n, both terms tend to zero under the stated convergence assumptions, so $b_{H,n} \to 0$. The temporal discounting bias does not enter this estimate because the comparison is anchored at the causal estimator.

More generally, if structural bias is not identically zero, the same argument gives

$$\|b_{H,n}\| \le \left\|b^{\mathrm{str}}_{CD,n}\right\| + K_T \varepsilon_{h,n} + C_\alpha C_G \bar{\mathcal{U}}_n.$$

Thus structural bias tending to zero still suffices for asymptotic unbiasedness, but it must be retained in finite-iteration bounds and rates. □

Corollary E.9.3 (Rates and disappearance of the averaged bias term)**.** *Under the assumptions of Theorem E.9.2, suppose there are deterministic constants $C_h, C_U < \infty$, exponents $p, q > 0$ and a deterministic index $n_0 \ge 1$ such that, almost surely for $n \ge n_0$, $\varepsilon_{h,n} \le C_h n^{-p}$ and $\bar{\mathcal{U}}_n \le C_U n^{-q}$. Let $r := \min\{p, q\}$. Then $\|b_{H,n}\| \le C n^{-r}$ almost surely for $n \ge n_0$ with $C := K_T C_h + C_\alpha C_G C_U$. For the practical mixing coefficient, Corollary E.5.7 gives $C_\alpha = \sqrt{\beta_{\max}}$. Moreover,*

$$\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}\,\|b_{H,n}\|^2 = \begin{cases} O(N^{-2r}), & 0 < r < \frac{1}{2}, \\ O(\log N/N), & r = \frac{1}{2}, \\ O(N^{-1}), & r > \frac{1}{2}. \end{cases} \tag{76}$$

The averaged squared bias therefore vanishes, and decays no slower than $O(N^{-1/2})$ whenever $p, q \ge 1/4$.

Proof. Let $r := \min\{p, q\}$. For $n \ge n_0$, Equation (75) and $n^{-p}, n^{-q} \le n^{-r}$ give, almost surely,

$$\begin{aligned}\|b_{H,n}\| &\le K_T C_h n^{-p} + C_\alpha C_G C_U n^{-q} \\ &\le (K_T C_h + C_\alpha C_G C_U) n^{-r} = C n^{-r}.\end{aligned}$$

Because C and n_0 are deterministic, squaring and taking expectations yields $\mathbb{E}\,\|b_{H,n}\|^2 \le C^2 n^{-2r}$ for every $n \ge n_0$. The initial sum $B_0 := \sum_{n=1}^{n_0-1} \mathbb{E}\,\|b_{H,n}\|^2$ is finite: Equation (72) gives $\|b_{H,n}\| \le \|b_R(\gamma)\| + C_G$, and Equation (24) bounds its right-hand side uniformly in n. For $N \ge n_0$,

$$\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}\,\|b_{H,n}\|^2 \le \frac{B_0}{N} + \frac{C^2}{N}\sum_{n=n_0}^{N} n^{-2r}.$$

Put $s := 2r > 0$. For the decreasing function x^{-s},

$$\sum_{n=1}^{N} n^{-s} \le 1 + \int_1^N x^{-s}\,dx = \begin{cases} 1 + (N^{1-s} - 1)/(1 - s), & 0 < s < 1, \\ 1 + \log N, & s = 1, \\ 1 + (1 - N^{1-s})/(s - 1), & s > 1. \end{cases}$$

Dividing by N gives the three cases in Equation (76); the finite-prefix term B_0/N respects each of those bounds. Each rate tends to zero. If $p, q \geq 1/4$, then $r \geq 1/4$: for $r < 1/2$, $N^{-2r} \leq N^{-1/2}$; for $r = 1/2$, $\log N/N = O(N^{-1/2})$; and for $r > 1/2$, $N^{-1} = O(N^{-1/2})$. For the practical mixing coefficient, Corollary E.5.7 supplies $C_\alpha = \sqrt{\beta_{\max}}$, as stated. □

F Delay-Conditioned Causal Credit

The pooled posterior of Equation (1) discards the delay. We retain the delay $Z := K_t - t$ as an additional hindsight input, giving the posterior and coefficient of Equation (4). Compared with the pooled construction, no new reward or policy-gradient term is introduced; only the hindsight posterior and its contribution coefficient are conditioned on the known delay. By Bayes' rule, $h^\star_{t,z}(a \mid s, u)/\pi_\theta(a \mid s) = p_{t,z}(u \mid s, a)/p_{t,z}(u \mid s)$, so $w^\star_{t,z}$ can be estimated by a delay-conditioned hindsight classifier exactly as in the pooled construction.

This section states precisely what delay conditioning buys, in four results. Appendix F.1 gives a strictly weaker unbiasedness requirement for non-reward-determining encodings. Appendix F.2 shows a strictly lower Bayes prediction risk when action and delay are conditionally nonseparable. Appendix F.3 gives an exact local oracle mean/variance comparison, identifying averaging as the mechanism common to the variance reduction pooling enjoys and the attenuation it suffers. Appendix F.4 proves a strict conditional fixed-design improvement in the detectability of localized causal influence.

F.1 Delay conditioning preserves the policy-gradient target

Assumption F.1.1 (Delay-conditioned reward sufficiency)**.** *For every admissible pair (t, z) there is an integrable measurable function $m_{t,z}$ such that, for almost every supported (s, a, u),*

$$\mathbb{E}[R_{t+z} \mid S_t = s, A_t = a, U_{t+z} = u] = m_{t,z}(u). \tag{77}$$

All ratios are evaluated only where $p_{t,z}(u \mid s) > 0$.

Lemma F.1.2 (Delay-conditioned advantage identity)**.** *Under Assumption F.1.1, for every supported (s, a, t, z),*

$$\mathbb{E}\big[w^\star_{t,z}(s, a, U_{t+z})R_{t+z} \mid S_t = s\big] = \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a] - \mathbb{E}[R_{t+z} \mid S_t = s]. \tag{78}$$

Proof. Abbreviate $p_a(u) := p_{t,z}(u \mid s, a)$, $p(u) := p_{t,z}(u \mid s)$, $m(u) := m_{t,z}(u)$. Averaging Equation (77) over A_t gives $\mathbb{E}[R_{t+z} \mid S_t = s, U_{t+z} = u] = m(u)$. Hence, by iterated expectation,

$$\begin{aligned}\mathbb{E}\Big[\frac{p_a(U_{t+z})}{p(U_{t+z})}R_{t+z} \;\Big|\; S_t = s\Big] &= \sum_{u:p(u)>0} p(u)\frac{p_a(u)}{p(u)}m(u) \\ &= \sum_u p_a(u)m(u) = \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a],\end{aligned} \tag{79}$$

where the last equality again follows from Equation (77). Subtracting $\mathbb{E}[R_{t+z} \mid S_t = s]$ and using $w^\star_{t,z} = p_a/p - 1$ proves Equation (78). For continuous U the same calculation uses densities and replaces sums by integrals. □

Theorem F.1.3 (Unbiased delay-conditioned causal policy gradient)**.** *Under Assumptions D.0.3 and F.1.1, define*

$$C^{\mathrm{CD}}_t(a) := \frac{\mathbf{1}\{A_t = a\}R_t}{\pi_\theta(a \mid S_t)} + \sum_{z=1}^{H_t} w^\star_{t,z}(S_t, a, U_{t+z})R_{t+z}, \quad G^\star_{CD} := \sum_{t=0}^{T-1}\sum_{a\in\mathcal{A}} \nabla_\theta \pi_\theta(a \mid S_t)C^{\mathrm{CD}}_t(a). \tag{80}$$

Then $\mathbb{E}[G^\star_{CD}] = \nabla_\theta J(\theta)$.

Proof. Define $Q^\pi_t(s, a) := \mathbb{E}[\sum_{k=t}^{T-1} R_k \mid S_t = s, A_t = a]$. For the immediate term,

$$\mathbb{E}\left[\frac{\mathbf{1}\{A_t = a\}R_t}{\pi_\theta(a \mid s)} \;\middle|\; S_t = s\right] = \mathbb{E}[R_t \mid S_t = s, A_t = a].$$

For each future delay, Lemma F.1.2 gives

$$\begin{aligned}
&\sum_a \nabla_\theta \pi_\theta(a \mid s)\mathbb{E}[w^\star_{t,z}(s,a,U_{t+z})R_{t+z} \mid S_t = s] \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s)\mathbb{E}[R_{t+z} \mid S_t = s, A_t = a] - \mathbb{E}[R_{t+z} \mid S_t = s]\sum_a \nabla_\theta \pi_\theta(a \mid s) \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s)\mathbb{E}[R_{t+z} \mid S_t = s, A_t = a],
\end{aligned}$$

since the last sum of policy derivatives is zero. Summing over delays and including the immediate term yields

$$\begin{aligned}
&\mathbb{E}\left[\sum_a \nabla_\theta \pi_\theta(a \mid S_t) C_t^{\mathrm{CD}}(a) \,\middle|\, S_t = s\right] \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s)\left\{\mathbb{E}[R_t \mid s, a] + \sum_{z=1}^{H_t} \mathbb{E}[R_{t+z} \mid s, a]\right\} \\
&= \sum_a \nabla_\theta \pi_\theta(a \mid s) Q_t^\pi(s,a).
\end{aligned}$$

For $t = T-1$, the future sum is empty. Taking expectations over the state at each source time and summing gives

$$\begin{aligned}
\mathbb{E}[G^\star_{CD}] &= \sum_{t=0}^{T-1} \mathbb{E}\left[\sum_a \nabla_\theta \pi_\theta(a \mid S_t) Q_t^\pi(S_t,a)\right] \\
&= \sum_{t=0}^{T-1} \mathbb{E}[\nabla_\theta \log \pi_\theta(A_t \mid S_t) Q_t^\pi(S_t,A_t)] \\
&= \sum_{t=0}^{T-1} \mathbb{E}\left[\nabla_\theta \log \pi_\theta(A_t \mid S_t) \sum_{k=t}^{T-1} R_k\right] = g.
\end{aligned}$$

Here the second equality uses the conditional action law π_θ, the third uses iterated expectation given (S_t, A_t), and the last uses the return-to-go identity proved in Proposition E.1.1. □

With the above theorem, it is important to ask when delay conditioning is necessary or helpful. If two source–outcome events are identical in all admissible non-delay variables, then no encoding restricted to those variables can distinguish them. In `Delay Dilution`, the informative and nuisance occurrences are generated by the same object and have the same observed outcome and reward. They therefore remain aliased under any encoding of the shared non-delay variables; the source–outcome delay Z is the distinguishing information. Thus, delay conditioning is necessary for this class of aliasing, though not for every failure caused by a coarse outcome encoding.

F.2 When delay provides strictly richer predictive information

Conditioning on the delay changes the learning problem. Consider predicting the source action A from the hindsight input $X = (t, S_t, U)$. The Bayes risk, the smallest expected cross-entropy loss any predictor can attain, is $H(A \mid X)$ for the pooled posterior $h^\star_t$ and $H(A \mid X, Z)$ for the delay-conditioned posterior $h^\star_{t,z}$. Delay conditioning therefore lowers it by exactly $I(A; Z \mid X)$, the information the delay carries about the action (Theorem F.2.1).

Formally, let X collect all non-delay inputs to the hindsight model, including the source-time index, source state, rewarding outcome and any fixed policy context, and write $h^\star_z(a \mid x) := \mathbb{P}(A = a \mid X = x, Z = z)$.

Theorem F.2.1 (Prediction-risk gain from delay information)**.** *Under logarithmic loss and unrestricted measurable categorical predictors, the Bayes risks of the pooled and delay-conditioned prediction problems are $\mathcal{L}^\star_{\mathrm{pool}} = H(A \mid X)$ and $\mathcal{L}^\star_{\mathrm{CD}} = H(A \mid X, Z)$. Consequently*

$$\mathcal{L}^\star_{\mathrm{pool}} - \mathcal{L}^\star_{\mathrm{CD}} = I(A; Z \mid X) \geq 0, \tag{81}$$

with strict improvement if and only if $A \not\perp Z \mid X$, that is, if the conditional action distribution changes with delay on a set of positive probability.

Proof. For any predictor $f(a \mid y)$ based on a conditioning variable Y, and each realized $Y = y$, adding and subtracting $-\sum_a p(a \mid y) \log p(a \mid y)$ gives

$$-\sum_a p(a \mid y) \log f(a \mid y) = H(A \mid Y = y) + \mathrm{KL}\big(p(\cdot \mid y) \,\|\, f(\cdot \mid y)\big). \tag{82}$$

The KL term is nonnegative and vanishes exactly at $f = p$ on the support, so $\inf_f \mathbb{E}[-\log f(A \mid Y)] = H(A \mid Y)$. Choosing $Y = X$ and $Y = (X, Z)$ and using $I(A; Z \mid X) = H(A \mid X) - H(A \mid X, Z)$ proves Equation (81). Conditional mutual information is zero exactly under conditional independence and strictly positive otherwise. □

The proposition below establishes the formal conditions under which delay conditioning can lead to a lower Bayes risk.

Proposition F.2.2 (Action-dependent temporal likelihood ratios)**.** *Fix (t, s, u) with the relevant probabilities positive. Under the exogenous delay-sampling convention of Section 3, for any actions a, b,*

$$\frac{\mathbb{P}(A_t = a \mid S_t = s, U_{t+z} = u, Z = z)}{\mathbb{P}(A_t = b \mid S_t = s, U_{t+z} = u, Z = z)} = \frac{\pi(a \mid s)}{\pi(b \mid s)} \cdot \frac{p_{t,z}(u \mid s, a)}{p_{t,z}(u \mid s, b)}. \tag{83}$$

Hence delay changes the conditional action distribution at (t, s, u) whenever there exist a, b, z_1, z_2 with

$$\frac{p_{t,z_1}(u \mid s, a)}{p_{t,z_1}(u \mid s, b)} \neq \frac{p_{t,z_2}(u \mid s, a)}{p_{t,z_2}(u \mid s, b)}. \tag{84}$$

If this nonseparability holds on a set of positive probability under the hindsight-prediction distribution, delay conditioning strictly reduces Bayes log loss by Theorem F.2.1. Conversely, if the likelihood factorizes as $p_{t,z}(u \mid s, a) = \tau_z(s, u)\, \kappa(s, a, u)$ for every supported s, u, a, z, then $A_t \perp Z \mid (t, S_t, U_{t+Z})$ and delay conditioning cannot reduce the Bayes prediction risk.

Proof. Bayes' rule gives $\mathbb{P}(A_t = a \mid s, u, z) = \pi(a \mid s) p_{t,z}(u \mid s, a) / p_{t,z}(u \mid s)$; writing the same expression for b and dividing cancels the normalizer, proving Equation (83). If Equation (84) holds, the posterior odds change with z, so the conditional action distribution is not constant in z; when such tuples have positive probability, conditional independence fails and Theorem F.2.1 gives strict improvement. Under the factorization, the ratio $\tau_z \kappa(s, a, u) / \{\tau_z \kappa(s, b, u)\} = \kappa(s, a, u) / \kappa(s, b, u)$ is independent of z; since a categorical distribution is determined by its odds relative to one reference action, the full posterior is independent of z. □

Equation (84) holds when actions induce different temporal response profiles for the same outcome: distinct first-passage-time distributions, path lengths, persistence times, bottleneck crossings, or delayed mechanisms. It fails when delay multiplies every action's outcome likelihood by a common factor. The relevant mathematical feature therefore is *nonseparability of action and delay in the outcome likelihood.*

F.3 Local oracle mean and variance under pooling and delay conditioning

We now make precise the claim stated in words in Section 3.2. The mathematical model below is the local abstraction of `Delay Dilution`: one rewarding object emits the same scalar reward at several delays, and only one delay carries action information.

Assumption F.3.1 (Localized causal delay with a shared outcome)**.** *Fix a source (t, s), an action a, and a single outcome value u whose associated reward is the constant $r \neq 0$. Let $N_z := \mathbf{1}\{U_{t+z} = u\}$ for $z \in \{1, \ldots, H_t\}$ be conditionally independent given $S_t = s$ with $p_z := p_{t,z}(u \mid s) \in [0, 1]$ and $p_{z^\star} \in (0, 1)$, and set $D := \sum_z p_z$. Suppose causal influence is localized at the delay $z^\star$ (the single informative delay):*

$$w^\star_{t,z^\star}(s, a, u) = w^\star_{z^\star} \neq 0, \quad w^\star_{t,z}(s, a, u) = 0 \ \ (z \neq z^\star), \quad \rho := \rho_{t,z^\star}(s, u) = \frac{p_{z^\star}}{D} \in (0, 1]. \tag{85}$$

Note that $\rho = 1$ requires $p_z = 0$ for all $z \neq z^\star$, i.e. a single supported delay; whenever at least two delays are supported, $\rho < 1$ strictly, which is the case of interest.

Proposition F.3.2 (Local oracle mean and variance comparison)**.** *Under Assumption F.3.1, define the pooled and delay-conditioned contributions at* (t, s, a)*,*

$$\Lambda_C := w_t^\star(s,a,u)\, r \sum_{z=1}^{H_t} N_z, \qquad \Lambda_{CD} := \sum_{z=1}^{H_t} w_{t,z}^\star(s,a,u)\, r N_z = w_{z^\star}^\star r N_{z^\star}, \tag{86}$$

where r *is a fixed nonzero reward associated with outcome* u*. Then* $w_t^\star(s,a,u) = \rho w_{z^\star}^\star$ *and*

$$\mathbb{E}[\Lambda_C \mid S_t = s] = \mathbb{E}[\Lambda_{CD} \mid S_t = s] = r\, p_{z^\star} w_{z^\star}^\star, \qquad \frac{\mathrm{Var}(\Lambda_C)}{\mathrm{Var}(\Lambda_{CD})} = \rho^2 \frac{\sum_z p_z(1-p_z)}{p_{z^\star}(1-p_{z^\star})}. \tag{87}$$

The ratio is smaller than one, i.e. pooling is less variable at equal mean, if and only if

$$p_{z^\star} \sum_z p_z(1-p_z) \;<\; (1-p_{z^\star})\, D^2. \tag{88}$$

In the homogeneous regime $p_z \equiv p$ *over* m *delays,* $\rho = 1/m$ *and the ratio is* $1/m$*; if the nuisance outcomes occur deterministically (*$p_z = 1$ *for* $z \neq z^\star$*) the ratio is* $\rho^2 < 1$*. In both of these regimes the pooled contribution is* less *variable than the delay-conditioned contribution at equal mean. Equation* (88) *can however fail: with* $H_t = 2$*,* $p_{z^\star} = 0.99$ *and* $p_2 = 0.01$ *the ratio is* ≈ 1.96*, so neither conditioning is uniformly less variable at the oracle level.*

Proof. By Lemma E.2.2 and Equation (85), $w_t^\star = \sum_z \rho_{t,z} w_{t,z}^\star = \rho w_{z^\star}^\star$. For the means, $\mathbb{E}[\Lambda_{CD} \mid S_t = s] = w_{z^\star}^\star r p_{z^\star}$ and

$$\mathbb{E}[\Lambda_C \mid S_t = s] = \rho w_{z^\star}^\star r \sum_z p_z = \rho w_{z^\star}^\star r D = \frac{p_{z^\star}}{D} w_{z^\star}^\star r D = w_{z^\star}^\star r p_{z^\star}.$$

This mean equality is not a coincidence: it is the local form of Theorem E.3.3, and it shows that attenuation by ρ is exactly compensated by summing over the ρ^{-1}-times larger expected number of occurrences D. For the variances, conditional independence of the N_z gives $\mathrm{Var}(\sum_z N_z) = \sum_z p_z(1-p_z)$ and $\mathrm{Var}(N_{z^\star}) = p_{z^\star}(1-p_{z^\star})$, so $\mathrm{Var}(\Lambda_C) = \rho^2 (w_{z^\star}^\star r)^2 \sum_z p_z(1-p_z)$ and $\mathrm{Var}(\Lambda_{CD}) = (w_{z^\star}^\star r)^2 p_{z^\star}(1-p_{z^\star})$. The stated ratio and its two specializations follow: for $p_z \equiv p$ over m delays, $\rho = p/(mp) = 1/m$ and $\sum_z p_z(1-p_z) = mp(1-p)$, giving $m^{-2} \cdot m = 1/m$; for deterministic nuisance outcomes $\sum_z p_z(1-p_z) = p_{z^\star}(1-p_{z^\star})$, giving ρ^2. Finally, substituting $\rho = p_{z^\star}/D$ into $\rho^2 \sum_z p_z(1-p_z) < p_{z^\star}(1-p_{z^\star})$ and cancelling one factor $p_{z^\star} > 0$ gives Equation (88); the numerical example follows by direct substitution, $D = 1$, $\rho = 0.99$, $\sum_z p_z(1-p_z) = 0.0198$ and $p_{z^\star}(1-p_{z^\star}) = 0.0099$. □

Note that the variance comparison in Proposition F.3.2 is meaningful because the pooled and delay-conditioned contributions have the same conditional mean. Comparisons obtained by rescaling the pooled coefficient or by treating the nuisance-delay contributions as zero-mean noise generally change that mean and therefore compare estimators of different quantities. Such comparisons cannot establish that one estimator is more efficient than the other for the original target.

In summary, summation reduces variance by averaging, and the ρ that makes the two means agree in Equation (87) is the ρ that shrinks the coefficient. An oracle with a known coefficient enjoys the variance reduction. A learner cannot: the quantity it estimates has been scaled by ρ while its sampling noise has not. Theorem F.4.1 quantifies this at the level of the learner and Theorem E.7.1 converts the resulting posterior error into gradient bias. The comparison here is between local oracle contributions under conditional independence and does not order the covariance of the full policy-gradient estimator.

F.4 What delay conditioning does buy: detectability of localized influence

The actual cost of pooling is statistical. The pooled target coefficient is $\rho w_{z^\star}^\star$, so a learner must resolve a deviation from the policy that is ρ times smaller. The comparison below conditions on a fixed delay design with $n_{z^\star} = n\rho$, thereby isolating the attenuation–splitting trade-off without mixing it with randomness in the delay count. Because the signal is attenuated linearly while SNR is quadratic in the signal, delay conditioning gains one asymptotic factor of ρ^{-1} despite using only a ρ fraction of the samples.

Theorem F.4.1 (Conditional fixed-design attenuation–splitting comparison)**.** *Adopt Assumption F.3.1. Fix an input $X = x = (t, s, u)$ and a delay design $Z_{1:n}$ containing exactly $n_{z^\star} = n\rho$ occurrences of the informative delay $z^\star$, where $n\rho$ is an integer. Conditional on this design, assume the action labels are independent, each distributed according to its delay-conditioned hindsight posterior. Fix the action a and write $\pi := \pi_\theta(a \mid s) \in (0, 1)$ and*

$$h^\star_{z^\star} := h^\star_{t,z^\star}(a \mid s, u) = \mathbb{P}(A = a \mid X = x, Z = z^\star) = \pi\big(1 + w^\star_{z^\star}\big) \in (0, 1). \tag{89}$$

For every nuisance delay $z \neq z^\star$, Assumption F.3.1 implies $\mathbb{P}(A = a \mid X = x, Z = z) = \pi$. Define

$$\widehat{h} := \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{A_i = a\}, \quad \widehat{h}_{z^\star} := \frac{1}{n\rho}\sum_{i:Z_i=z^\star} \mathbf{1}\{A_i = a\}, \quad \widehat{w} := \frac{\widehat{h}}{\pi} - 1, \quad \widehat{w}_{z^\star} := \frac{\widehat{h}_{z^\star}}{\pi} - 1. \tag{90}$$

Then, conditional on $Z_{1:n}$,

$$\mathbb{E}[\widehat{w} \mid Z_{1:n}] = \rho w^\star_{z^\star}, \qquad \mathrm{Var}(\widehat{w} \mid Z_{1:n}) = \frac{\rho h^\star_{z^\star}(1 - h^\star_{z^\star}) + (1 - \rho)\pi(1 - \pi)}{\pi^2 n}, \tag{91}$$

$$\mathbb{E}[\widehat{w}_{z^\star} \mid Z_{1:n}] = w^\star_{z^\star}, \qquad \mathrm{Var}(\widehat{w}_{z^\star} \mid Z_{1:n}) = \frac{h^\star_{z^\star}(1 - h^\star_{z^\star})}{\pi^2 n\rho}. \tag{92}$$

With $\mathrm{SNR}_{\mathrm{D}}(\widehat{w}) := (\mathbb{E}[\widehat{w} \mid Z_{1:n}])^2 / \mathrm{Var}(\widehat{w} \mid Z_{1:n})$,

$$\frac{\mathrm{SNR}_{\mathrm{D}}(\widehat{w}_{z^\star})}{\mathrm{SNR}_{\mathrm{D}}(\widehat{w})} = 1 + \frac{(1 - \rho)\pi(1 - \pi)}{\rho h^\star_{z^\star}(1 - h^\star_{z^\star})}. \tag{93}$$

The ratio is strictly greater than one if and only if $\rho < 1$, and equals one when $\rho = 1$. Along a family with π and $h^\star_{z^\star}$ fixed in $(0, 1)$ and $n\rho$ a positive integer, it is $\Theta(\rho^{-1})$ as $\rho \to 0$.

Proof. By Bayes' rule, $h^\star_{t,z}(a \mid s, u) = \pi\{1 + w^\star_{t,z}(s, a, u)\}$. Hence the informative-delay success probability is $h^\star_{z^\star} = \pi(1 + w^\star_{z^\star})$, while every nuisance-delay success probability is π because its coefficient is zero.

Conditional on the fixed design, the $n\rho$ informative labels are independent Bernoulli$(h^\star_{z^\star})$ and the remaining $n(1 - \rho)$ labels are independent Bernoulli(π). Therefore

$$\mathbb{E}[\widehat{h} \mid Z_{1:n}] = \rho h^\star_{z^\star} + (1 - \rho)\pi = \pi(1 + \rho w^\star_{z^\star}), \tag{94}$$

and independence gives

$$\mathrm{Var}(\widehat{h} \mid Z_{1:n}) = \frac{\rho h^\star_{z^\star}(1 - h^\star_{z^\star}) + (1 - \rho)\pi(1 - \pi)}{n}. \tag{95}$$

Dividing by π and subtracting one proves Equation (91). The delay-conditioned estimator is the mean of $n\rho$ independent Bernoulli$(h^\star_{z^\star})$ labels, which proves Equation (92).

Substituting these moments into the conditional SNRs and cancelling the common factor $(w^\star_{z^\star})^2\pi^2 n$ yields

$$\frac{\mathrm{SNR}_{\mathrm{D}}(\widehat{w}_{z^\star})}{\mathrm{SNR}_{\mathrm{D}}(\widehat{w})} = \frac{\rho h^\star_{z^\star}(1 - h^\star_{z^\star}) + (1 - \rho)\pi(1 - \pi)}{\rho h^\star_{z^\star}(1 - h^\star_{z^\star})} \tag{96}$$

$$= 1 + \frac{(1 - \rho)\pi(1 - \pi)}{\rho h^\star_{z^\star}(1 - h^\star_{z^\star})}. \tag{97}$$

All probabilities in the denominator lie in $(0, 1)$, so the added term is positive exactly when $\rho < 1$ and vanishes at $\rho = 1$. With $c := \pi(1 - \pi)/\{h^\star_{z^\star}(1 - h^\star_{z^\star})\} > 0$ fixed, the ratio is $1 + c(1 - \rho)/\rho$. Multiplying by ρ gives $\rho + c(1 - \rho) \to c > 0$, proving the stated $\Theta(\rho^{-1})$ order along the specified family. □

Remark F.4.2 (Scope of Theorem F.4.1). The theorem is a conditional fixed-design statement about the hindsight-learning problem, not a covariance ordering for the full MDP gradient. It is also, deliberately, not an efficiency comparison between two estimators of a common target: $\widehat{w}_{z^\star}$ estimates $w^\star_{z^\star}$, whereas $\widehat{w}$ estimates $\rho w^\star_{z^\star}$. The ratio Equation (93) therefore compares how many standard errors each places between its estimate and the value "no contribution", which is the quantity a learner must resolve before the coefficient can influence the update. This is why it does not contradict the same-target requirement stressed after Proposition F.3.2: that requirement governs variance orderings, whereas the present statement is about detectability of a nonzero contribution. Conditioning

on $n_{z^\star} = n\rho$ avoids the zero-count event and the additional variance caused by a random binomial delay count. Under i.i.d. random delay sampling, $n_{z^\star}/n \to \rho$ for fixed $\rho > 0$. In a simultaneous rare-delay limit, informative-count concentration additionally requires $n\rho \to \infty$. With this coverage and fixed nondegenerate π and $h^\star_{z^\star}$, the inverse-ρ order is also obtained asymptotically; the unconditional finite-sample ratio need not equal Equation (93).

The policy-gradient relevance comes from Theorem E.7.1: posterior error enters the causal-gradient bias linearly, so a coefficient that the learner cannot resolve contributes error rather than useful credit. In particular, a learner with an additive error floor δ resolves $w^\star_{z^\star}$ whenever $\delta < \pi\,|w^\star_{z^\star}|$, but resolves $\rho w^\star_{z^\star}$ only when $\delta < \pi\rho\,|w^\star_{z^\star}|$.

The following proposition makes precise the error-floor statement in Section 3.2 and Remark F.4.2. We model a learner with an additive error floor δ as one whose posterior estimate is accurate only up to an absolute error δ. We then ask when such an estimate is guaranteed to recover the sign of the contribution, and in particular to separate it from zero.

Proposition F.4.3 (Sign recovery under an additive posterior error floor)**.** *Adopt Assumption F.3.1 and write $\pi := \pi_\theta(a \mid s) \in (0,1)$. Fix $\delta \geq 0$. Let $\widetilde{h}_{z^\star}$ and $\widetilde{h}$ be any estimates of the delay-conditioned posterior $h^\star_{t,z^\star}(a \mid s,u)$ and the pooled posterior $h^\star_t(a \mid s,u)$ whose errors are at most δ,*

$$\left|\widetilde{h}_{z^\star} - h^\star_{t,z^\star}(a \mid s,u)\right| \leq \delta, \qquad \left|\widetilde{h} - h^\star_t(a \mid s,u)\right| \leq \delta, \tag{98}$$

and let $\widetilde{w}_{z^\star} := \widetilde{h}_{z^\star}/\pi - 1$ and $\widetilde{w} := \widetilde{h}/\pi - 1$ be the corresponding contribution coefficients. Then:

(i) $\operatorname{sign}(\widetilde{w}_{z^\star}) = \operatorname{sign}(w^\star_{z^\star})$ *for every $\widetilde{h}_{z^\star}$ satisfying Equation* (98) *if and only if $\delta < \pi\,|w^\star_{z^\star}|$;*

(ii) $\operatorname{sign}(\widetilde{w}) = \operatorname{sign}(w^\star_{z^\star})$ *for every $\widetilde{h}$ satisfying Equation* (98) *if and only if $\delta < \pi\rho\,|w^\star_{z^\star}|$.*

Consequently, if $\rho < 1$ and $\pi\rho\,|w^\star_{z^\star}| \leq \delta < \pi\,|w^\star_{z^\star}|$, every delay-conditioned estimate within the error floor has the correct sign, whereas some pooled estimate within the same floor gives $\widetilde{w} = 0$, which is indistinguishable from no contribution.

Proof. We first prove a general fact. Let $h^\star \in [0,1]$ be a posterior with $\chi := h^\star - \pi \neq 0$, let $\widetilde{h}$ satisfy $\left|\widetilde{h} - h^\star\right| \leq \delta$, and set $\widetilde{w} := \widetilde{h}/\pi - 1 = (\widetilde{h} - \pi)/\pi$. Since $\pi > 0$, $\operatorname{sign}(\widetilde{w}) = \operatorname{sign}(\widetilde{h} - \pi)$, and

$$\widetilde{h} - \pi = \chi + \big(\widetilde{h} - h^\star\big).$$

If $\delta < |\chi|$, then $\left|\widetilde{h} - h^\star\right| \leq \delta < |\chi|$, so $\widetilde{h} - \pi$ has the sign of χ for every admissible $\widetilde{h}$. If $\delta \geq |\chi|$, the choice $\widetilde{h} := \pi$ is a valid probability, satisfies $\left|\widetilde{h} - h^\star\right| = |\chi| \leq \delta$, and gives $\widetilde{w} = 0$, whose sign differs from that of χ. Hence

$$\operatorname{sign}(\widetilde{w}) = \operatorname{sign}(\chi) \ \text{ for every admissible } \widetilde{h} \quad \Longleftrightarrow \quad \delta < |\chi|\,. \tag{99}$$

It remains to compute χ for the two posteriors. For the delay-conditioned posterior, Equation (4) gives $h^\star_{t,z^\star}(a \mid s,u) = \pi(1 + w^\star_{z^\star})$, so $\chi = \pi w^\star_{z^\star}$. For the pooled posterior, Equation (1) gives $h^\star_t(a \mid s,u) = \pi\{1 + w^\star_t(s,a,u)\}$, and Lemma E.2.2 together with Equation (85) gives $w^\star_t(s,a,u) = \sum_z \rho_{t,z}(s,u)\, w^\star_{t,z}(s,a,u) = \rho\, w^\star_{z^\star}$; hence $\chi = \pi\rho\, w^\star_{z^\star}$. In both cases $\chi \neq 0$ and $\operatorname{sign}(\chi) = \operatorname{sign}(w^\star_{z^\star})$, because $\pi > 0$, $\rho = p_{z^\star}/D > 0$ under Assumption F.3.1, and $w^\star_{z^\star} \neq 0$. Applying Equation (99) with $|\chi| = \pi\,|w^\star_{z^\star}|$ proves (i), and with $|\chi| = \pi\rho\,|w^\star_{z^\star}|$ proves (ii). For the final statement, $\delta < \pi\,|w^\star_{z^\star}|$ gives the correct sign for every delay-conditioned estimate by (i), while $\delta \geq \pi\rho\,|w^\star_{z^\star}| = |\chi|$ for the pooled posterior, so the choice $\widetilde{h} = \pi$ from the first paragraph gives $\widetilde{w} = 0$; the interval for δ is nonempty because $\rho < 1$. □

The fixed-design theorem prices delay conditioning at $n\rho$ informative samples. As we will see in Appendix G, kernel smoothing can raise the effective sample size to $n_{\text{eff}} \geq n\rho$, but it changes the target from the delay-conditioned posterior $h^\star_{t,z^\star}$ to a local mixture and therefore introduces smoothing bias. Theorem G.4.1 gives a sufficient condition for lower conditional MSE.

G KERNEL-SMOOTHED DELAY CONDITIONING

Delay conditioning resolves temporal aliasing but splits supervision across delays: the effective sample size at the informative delay is $n\rho$, which is exactly the quantity that limits the gain in Theorem F.4.1. The kernelized method addresses this sparsity by letting an observation generated at delay z' contribute to the loss evaluated at a nearby query delay z. Throughout, X collects the non-delay hindsight inputs, $p_z(a \mid x) := \mathbb{P}(A = a \mid X = x, Z = z)$, the bandwidth is $\sigma > 0$, and $K_\sigma(z, z') \geq 0$ is a kernel with $K_\sigma(z, z) > 0$. The training objective is the normalized kernel-weighted cross-entropy $\widetilde{\mathcal{L}}_\sigma$ of Equation (11), whose weights

$$\omega_\sigma(z' \mid x, z) := \frac{p(z' \mid x)\, K_\sigma(z, z')}{D_\sigma(x, z)}, \qquad D_\sigma(x, z) := \sum_{z'} p(z' \mid x)\, K_\sigma(z, z'), \tag{100}$$

are nonnegative and sum to one; we assume $D_\sigma(x, z) > 0$ at every query.

Assumption G.0.1 (Kernel family)**.** *$K_\sigma(z, z') \geq 0$, $K_\sigma(z, z) > 0$, and the family is (i)* concentrating*: $K_\sigma(z, z')/K_\sigma(z, z) \to 0$ as $\sigma \downarrow 0$ for every $z' \neq z$; and (ii)* flattening*: $K_\sigma(z, z')/K_\sigma(z, z) \to 1$ as $\sigma \to \infty$ for every z' in the (finite) delay range. The Gaussian kernel $K_\sigma(z, z') = \exp\{-(z - z')^2/2\sigma^2\}$ used in the implementation satisfies both. Only these two limits are used, and only in Remark G.2.1; every other statement in this appendix holds for an arbitrary nonnegative kernel with $K_\sigma(z, z) > 0$.*

This appendix establishes four facts about that objective. It targets a convex mixture of neighboring delay-conditioned posteriors (Appendix G.1). The bandwidth interpolates between delay-conditioned and delay-pooled estimators (Appendix G.2). Smoothing introduces bias but can raise the effective sample size (Appendix G.3). Finally, smoothing lowers the conditional MSE exactly when the variance saved exceeds the squared bias incurred (Appendix G.4).

Theorem G.0.2 (Bandwidth interpolation between delay-conditioned and delay-pooled estimators)**.** *For a fixed query (x, z), the population minimizer of Equation* (11) *is the convex combination of delay-conditioned posteriors*

$$p_\sigma(\cdot \mid x, z) = \sum_{z'} \omega_\sigma(z' \mid x, z)\, p_{z'}(\cdot \mid x). \tag{101}$$

Suppose $p(z \mid x) > 0$ and the kernel family concentrates as $\sigma \downarrow 0$ and flattens as $\sigma \to \infty$ (Assumption G.0.1). Then $p_\sigma(\cdot \mid x, z) \to p_z(\cdot \mid x)$ as $\sigma \downarrow 0$, while $p_\sigma(\cdot \mid x, z) \to p(\cdot \mid x)$ as $\sigma \to \infty$, where $p(\cdot \mid x)$ is the pooled posterior. Proof: Proposition G.1.1 and Remark G.2.1.

G.1 POPULATION TARGET OF THE KERNELIZED LOSS

Proposition G.1.1 (Population target of kernelized cross-entropy)**.** *Let $\widetilde{\mathcal{L}}_\sigma$ be as in Equation* (11) *with the weights of Equation* (100) *and $D_\sigma(x, z) > 0$. Then $\widetilde{\mathcal{L}}_\sigma(\cdot\,; x, z)$ is minimized over categorical distributions, uniquely on the support of the minimizer, by*

$$p_\sigma(\cdot \mid x, z) = \sum_{z'} \omega_\sigma(z' \mid x, z)\, p_{z'}(\cdot \mid x). \tag{102}$$

Positive-bandwidth training therefore targets a convex mixture of neighboring delay-conditioned posteriors rather than the delay-conditioned posterior p_z.

Proof. Fix (x, z) and a categorical candidate $f(\cdot \mid x, z)$. Expanding the conditional loss and exchanging the finite sums gives

$$\begin{aligned}
\widetilde{\mathcal{L}}_\sigma(f; x, z) &= \sum_{z'} \omega_\sigma(z' \mid x, z) \left[-\sum_a p_{z'}(a \mid x) \log f(a \mid x, z) \right] \\
&= -\sum_a \left[\sum_{z'} \omega_\sigma(z' \mid x, z) p_{z'}(a \mid x) \right] \log f(a \mid x, z) \\
&= -\sum_a p_\sigma(a \mid x, z) \log f(a \mid x, z).
\end{aligned}$$

The proposed mixture is nonnegative and satisfies

$$\sum_a p_\sigma(a \mid x, z) = \sum_{z'} \omega_\sigma(z' \mid x, z) \sum_a p_{z'}(a \mid x) = \sum_{z'} \omega_\sigma(z' \mid x, z) = 1.$$

Adding and subtracting its entropy gives

$$\widetilde{\mathcal{L}}_\sigma(f; x, z) = H(p_\sigma(\cdot \mid x, z)) + \mathrm{KL}(p_\sigma(\cdot \mid x, z) \,\|\, f(\cdot \mid x, z)).$$

If f vanishes where p_σ is positive, the loss is infinite. Otherwise the KL term is nonnegative, with equality exactly when $f = p_\sigma$ as distributions. The entropy does not depend on f, so the mixture in Equation (102) attains the minimum. □

Note that Proposition G.1.1 requires the bandwidth to be independent of the action a, because K_σ is factored out of the sum over actions. An action-dependent bandwidth would instead induce the per-action reweighting $m_a \propto \sum_{z'} p(z' \mid x) K_{\sigma(a)}(z, z') p_{z'}(a \mid x)$, whose normalization is action dependent and which is not a mixture of delay-conditioned posteriors. This is why the bandwidth head is conditioned on (x, z) only.

G.2 The bandwidth interpolates between delay-conditioned and delay-pooled estimators

Remark G.2.1 (Bandwidth endpoints, and a bias bound uniform in σ). Fix a supported query $x = (t, s, u), z$ with $p(z \mid x) > 0$. By Lemma E.2.3, $p(z' \mid x) = \rho_{t,z'}(s, u)$. Set $r_\sigma(z') := K_\sigma(z, z')/K_\sigma(z, z)$, so $r_\sigma(z) = 1$ and

$$\omega_\sigma(z' \mid x, z) = \frac{p(z' \mid x) r_\sigma(z')}{p(z \mid x) + \sum_{j \neq z} p(j \mid x) r_\sigma(j)}.$$

For $\sigma \downarrow 0$, Assumption G.0.1 gives $r_\sigma(z') \to 0$ for $z' \neq z$. Since the delay range is finite,

$$\omega_\sigma(z' \mid x, z) \longrightarrow \mathbf{1}\{z' = z\}, \qquad p_\sigma(\cdot \mid x, z) \longrightarrow p_z(\cdot \mid x).$$

For $\sigma \to \infty$, all $r_\sigma(z') \to 1$, and hence

$$\omega_\sigma(z' \mid x, z) \longrightarrow p(z' \mid x), \qquad p_\sigma(\cdot \mid x, z) \longrightarrow \sum_{z'} p(z' \mid x) p_{z'}(\cdot \mid x) = p(\cdot \mid x) = h^\star_t(\cdot \mid s, u).$$

The last equality is the conditional law of total probability. Thus the endpoints recover delay-conditioned and pooled hindsight, respectively.

For every finite $\sigma > 0$, write $\mathcal{Z}_x := \{z' : p(z' \mid x) > 0\}$ and suppress the fixed arguments. Because the weights sum to one,

$$p_\sigma - p_z = \sum_{z' \in \mathcal{Z}_x} \omega_\sigma(z')(p_{z'} - p_z).$$

The triangle inequality and nonnegative weights give

$$\begin{aligned} \|p_\sigma - p_z\|_1 &\leq \sum_{z' \in \mathcal{Z}_x} \omega_\sigma(z') \, \|p_{z'} - p_z\|_1 \\ &\leq \max_{z' \in \mathcal{Z}_x} \|p_{z'} - p_z\|_1 =: D(x, z) \leq 2. \end{aligned} \tag{103}$$

The last inequality holds because both vectors are probability distributions. No delay-smoothness assumption is needed. Under Assumption D.0.4, Theorem E.7.1 and Remark E.7.2 therefore bound the additional causal-gradient bias by $K_T \mathbb{E}[D(X, Z)] \leq 2K_T$. This bias arises from replacing the delay-conditioned posterior with this mixture, and the expectation uses the on-policy uniformly sampled pair distribution. This bound is uniform in bandwidth and concerns the smoothing contribution to bias.

G.3 Smoothing bias and effective sample size

Smoothing moves the target away from p_z, which costs bias, and pools observations from neighboring delays, which reduces variance. We quantify the two separately here and trade them off in Appendix G.4.

Assumption G.3.1 (Local delay smoothness)**.** *For some exponent $\gamma_b > 0$ and a measurable $L(x) < \infty$, $\|p_{z'}(\cdot \mid x) - p_z(\cdot \mid x)\|_1 \le L(x)\,|z' - z|^{\gamma_b}$ for all delays receiving nonzero kernel weight at the query (x, z).*

Proposition G.3.2 (Kernel smoothing bias)**.** *Under Assumption G.3.1,*

$$\|p_\sigma(\cdot \mid x, z) - p_z(\cdot \mid x)\|_1 \le L(x) \sum_{z'} \omega_\sigma(z' \mid x, z)|z' - z|^{\gamma_b}.$$

If the normalized kernel satisfies $\sum_{z'} \omega_\sigma(z' \mid x, z)|z' - z|^{\gamma_b} \le C_K \sigma^{\gamma_b}$, then

$$\|p_\sigma(\cdot \mid x, z) - p_z(\cdot \mid x)\|_1 \le L(x) C_K \sigma^{\gamma_b}. \tag{104}$$

Assume also Assumption D.0.4, and let expectations below use the on-policy uniformly sampled source–delay pair distribution. For a common deterministic bandwidth and $\mathbb{E}[L(X)] < \infty$, the bias contributed by smoothing the delay-conditioned posterior satisfies

$$\left\|b_G^{\mathrm{smooth}}(\sigma)\right\| \le K_T C_K \mathbb{E}[L(X)]\, \sigma^{\gamma_b}.$$

For a measurable query-dependent bandwidth $\sigma(X, Z)$, with the same moment bound at every query, the corresponding statement is

$$\left\|b_G^{\mathrm{smooth}}\right\| \le K_T C_K \mathbb{E}[L(X)\sigma(X, Z)^{\gamma_b}].$$

In particular, under changing on-policy laws, a sufficient condition for the smoothing bias to vanish is

$$\sup_n \mathbb{E}[L(X_n)] < \infty, \qquad \sigma_n(X_n, Z_n) \le b_n \quad \textit{almost surely},$$

where $b_n \to 0$ is deterministic and C_K is independent of n.

Proof. Fix (x, z) and suppress the common conditioning arguments. From Equation (102) and $\sum_{z'} \omega_\sigma(z') = 1$,

$$p_\sigma - p_z = \sum_{z'} \omega_\sigma(z')(p_{z'} - p_z).$$

Therefore

$$\begin{aligned}
\|p_\sigma - p_z\|_1 &= \sum_a \left| \sum_{z'} \omega_\sigma(z')(p_{z'}(a) - p_z(a)) \right| \\
&\le \sum_{z'} \omega_\sigma(z') \sum_a |p_{z'}(a) - p_z(a)| \\
&= \sum_{z'} \omega_\sigma(z')\, \|p_{z'} - p_z\|_1 \\
&\le L(x) \sum_{z'} \omega_\sigma(z')|z' - z|^{\gamma_b} \le L(x) C_K \sigma^{\gamma_b}.
\end{aligned}$$

The penultimate inequality is Assumption G.3.1; the last uses the additional kernel moment hypothesis. This proves both pointwise claims.

At the population kernel target, Theorem E.7.1 and Remark E.7.2 apply to the difference between the smoothed and delay-conditioned estimators. For a common deterministic bandwidth,

$$\begin{aligned}
\left\|b_G^{\mathrm{smooth}}(\sigma)\right\| &\le K_T \mathbb{E}\, \|p_\sigma(\cdot \mid X, Z) - p_Z(\cdot \mid X)\|_1 \\
&\le K_T C_K \mathbb{E}[L(X)\sigma^{\gamma_b}] = K_T C_K \mathbb{E}[L(X)]\sigma^{\gamma_b}.
\end{aligned}$$

For a query-dependent bandwidth the same pointwise calculation instead gives

$$\left\|b_G^{\mathrm{smooth}}\right\| \le K_T C_K \mathbb{E}[L(X)\sigma(X, Z)^{\gamma_b}],$$

with the bandwidth retained inside the expectation. Finally, if $\sigma_n(X_n, Z_n) \le b_n$ and $L_* := \sup_n \mathbb{E}[L(X_n)] < \infty$, then

$$\left\|b_{G,n}^{\mathrm{smooth}}\right\| \le K_T C_K L_* b_n^{\gamma_b} \longrightarrow 0.$$

These bounds concern only the bias added by smoothing, independently of any structural bias of the exact oracle or estimation error of a learned model. □

For the variance side, fix (x, z, a), let $Y_i := \mathbf{1}\{A_i = a\}$, let Z_i be the observed delays, and define the empirical weights, the kernel estimator, and the effective sample size

$$\omega_i := \frac{K_\sigma(z, Z_i)}{\sum_{j=1}^n K_\sigma(z, Z_j)}, \quad \widehat{p}_\sigma(a \mid x, z) := \sum_{i=1}^n \omega_i Y_i, \quad n_{\text{eff}} := \Big(\sum_i \omega_i^2\Big)^{-1}. \tag{105}$$

For comparison, $\widehat{p}_z(a \mid x) := n_z^{-1} \sum_{i:Z_i=z} Y_i$ denotes the delay-conditioned estimator, which uses only the $n_z := |\{i : Z_i = z\}|$ observations at delay z.

Proposition G.3.3 (Conditional variance and effective sample size)**.** *Condition on $(Z_{1:n}, X = x)$ and prescribe the bandwidth independently of the labels used below, so that the weights in Equation* (105) *are fixed with positive denominator. Assume that, under this conditioning, the Y_i are independent Bernoulli variables with success probabilities $p_{Z_i}(a \mid x)$. Then*

$$\mathrm{Var}\big(\widehat{p}_\sigma \mid Z_{1:n}, X = x\big) = \sum_i \omega_i^2\, p_{Z_i}(a \mid x)\{1 - p_{Z_i}(a \mid x)\} \le \frac{1}{4n_{\text{eff}}}. \tag{106}$$

If the posterior is locally homogeneous on the kernel support, i.e. $p_{Z_i}(a \mid x) = p_z(a \mid x)$ whenever $\omega_i > 0$, then the variance equals $p_z(a \mid x)\{1 - p_z(a \mid x)\}/n_{\text{eff}}$, and with $n_z \ge 1$ observations at delay z the relative variance change against the delay-conditioned estimator is $1 - n_z/n_{\text{eff}}$ when $0 < p_z(a \mid x) < 1$. If $p_z(a \mid x) \in \{0, 1\}$, both variances are zero under local homogeneity and no relative ratio is defined. If in addition the kernel is maximized at the query delay, $0 < K_\sigma(z, z)$ and $K_\sigma(z, z') \le K_\sigma(z, z)$ for every observed z', then $n_{\text{eff}} \ge n_z$, so the change is a genuine nonnegative reduction.

Proof. Write $\mathcal{D}$ for the fixed conditioning information, including $(Z_{1:n}, X = x)$ and the prescribed bandwidth. The weights are deterministic under this conditioning. Thus

$$\begin{aligned}
\mathrm{Var}(\widehat{p}_\sigma \mid \mathcal{D}) &= \sum_i \omega_i^2 \,\mathrm{Var}(Y_i \mid \mathcal{D}) + 2\sum_{i<j} \omega_i \omega_j \,\mathrm{Cov}(Y_i, Y_j \mid \mathcal{D}) \\
&= \sum_i \omega_i^2 p_{Z_i}(a \mid x)\{1 - p_{Z_i}(a \mid x)\} \\
&\le \frac{1}{4}\sum_i \omega_i^2 = \frac{1}{4n_{\text{eff}}}.
\end{aligned}$$

Conditional independence removes the cross covariances, and $p(1-p) \le 1/4$ gives the inequality. Under local homogeneity, writing $p_z := p_z(a \mid x)$, the two conditional variances are

$$\mathrm{Var}(\widehat{p}_\sigma \mid \mathcal{D}) = \frac{p_z(1-p_z)}{n_{\text{eff}}}, \qquad \mathrm{Var}(\widehat{p}_z \mid \mathcal{D}) = \frac{p_z(1-p_z)}{n_z}.$$

When $0 < p_z < 1$, the fraction of the delay-conditioned variance removed is

$$\frac{\mathrm{Var}(\widehat{p}_z \mid \mathcal{D}) - \mathrm{Var}(\widehat{p}_\sigma \mid \mathcal{D})}{\mathrm{Var}(\widehat{p}_z \mid \mathcal{D})} = 1 - \frac{n_z}{n_{\text{eff}}}.$$

When $p_z \in \{0, 1\}$ both variances vanish.

For the final claim, put $k_i := K_\sigma(z, Z_i)$, $c := K_\sigma(z, z) > 0$, and $S := \sum_i k_i$. Since $0 \le k_i \le c$, $\sum_i k_i^2 \le cS$, whereas $S \ge n_z c > 0$. Consequently,

$$n_{\text{eff}} = \frac{S^2}{\sum_i k_i^2} \ge \frac{S}{c} \ge n_z.$$

If an observation from another delay receives positive weight, then $S > n_z c$ and the final inequality is strict. □

G.4 When smoothing helps: variance reduction against squared bias

Theorem G.4.1 (Finite-sample condition for beneficial smoothing)**.** *For one posterior component $p_z := p_z(a \mid x)$, a bandwidth fixed independently of the labels used for estimation, and a fixed*

realized delay design with $n_z \geq 1$, suppose that for constants $C_b, C_v > 0$ and a verified bias exponent $\gamma_b > 0$,

$$\begin{aligned} &\left|\mathbb{E}[\widehat{p}_\sigma \mid Z_{1:n}, X = x] - p_z\right| \leq C_b \sigma^{\gamma_b}, \\ &\mathrm{Var}(\widehat{p}_\sigma \mid Z_{1:n}, X = x) \leq \frac{C_v}{n_{\mathrm{eff}}}, \qquad \mathrm{Var}(\widehat{p}_z \mid Z_{1:n}, X = x) = \frac{C_v}{n_z}, \end{aligned} \tag{107}$$

with the delay-conditioned estimator conditionally unbiased. Then $\mathrm{MSE}(\widehat{p}_\sigma) \leq C_b^2 \sigma^{2\gamma_b} + C_v / n_{\mathrm{eff}}$, and a sufficient condition for the kernel estimator to have strictly lower conditional MSE than the delay-conditioned estimator is

$$C_b^2 \sigma^{2\gamma_b} < C_v \Big(\frac{1}{n_z} - \frac{1}{n_{\mathrm{eff}}} \Big). \tag{108}$$

Proof. Condition throughout on $\mathcal{D} := (Z_{1:n}, X = x)$, with the bandwidth fixed as in the conditional analysis. Let $m_\sigma := \mathbb{E}[\widehat{p}_\sigma \mid \mathcal{D}]$ and $b_\sigma := m_\sigma - p_z$. Since $\mathbb{E}[\widehat{p}_\sigma - m_\sigma \mid \mathcal{D}] = 0$,

$$\begin{aligned} \mathrm{MSE}(\widehat{p}_\sigma \mid \mathcal{D}) &:= \mathbb{E}[(\widehat{p}_\sigma - p_z)^2 \mid \mathcal{D}] \\ &= \mathbb{E}[(\widehat{p}_\sigma - m_\sigma + b_\sigma)^2 \mid \mathcal{D}] \\ &= \mathrm{Var}(\widehat{p}_\sigma \mid \mathcal{D}) + 2 b_\sigma \mathbb{E}[\widehat{p}_\sigma - m_\sigma \mid \mathcal{D}] + b_\sigma^2 \\ &= \mathrm{Var}(\widehat{p}_\sigma \mid \mathcal{D}) + b_\sigma^2 \\ &\leq \frac{C_v}{n_{\mathrm{eff}}} + C_b^2 \sigma^{2\gamma_b}. \end{aligned}$$

For the delay-conditioned estimator, conditional unbiasedness gives

$$\begin{aligned} \mathrm{MSE}(\widehat{p}_z \mid \mathcal{D}) &:= \mathbb{E}[(\widehat{p}_z - p_z)^2 \mid \mathcal{D}] \\ &= \mathrm{Var}(\widehat{p}_z \mid \mathcal{D}) + (\mathbb{E}[\widehat{p}_z \mid \mathcal{D}] - p_z)^2 = \frac{C_v}{n_z}. \end{aligned}$$

Subtracting yields

$$\mathrm{MSE}(\widehat{p}_\sigma \mid \mathcal{D}) - \mathrm{MSE}(\widehat{p}_z \mid \mathcal{D}) \leq C_b^2 \sigma^{2\gamma_b} - C_v \left(\frac{1}{n_z} - \frac{1}{n_{\mathrm{eff}}} \right).$$

The right-hand side is strictly negative under Equation (108), proving the claimed sufficient condition. Necessarily this sufficient condition requires $n_{\mathrm{eff}} > n_z$; when it fails, the upper bounds alone do not determine the MSE ordering. □

Remark G.4.2 (The bias hypothesis is the conditional form of Proposition G.3.2). The population target uses weights proportional to $p(z' \mid x) K_\sigma(z, z')$, whereas the conditional mean of the sample estimator uses observed counts. For a fixed design and bandwidth, let $n_{z'} := \sum_i \mathbf{1}\{Z_i = z'\}$ and

$$\widehat{\omega}(z') := \frac{n_{z'} K_\sigma(z, z')}{\sum_j n_j K_\sigma(z, j)}.$$

Under the conditional label law of Proposition G.3.3,

$$\begin{aligned} \mathbb{E}[\widehat{p}_\sigma(a \mid x, z) \mid Z_{1:n}, X = x] &= \sum_i \omega_i p_{Z_i}(a \mid x) \\ &= \sum_{z'} \widehat{\omega}(z') p_{z'}(a \mid x). \end{aligned}$$

Since these weights are nonnegative and sum to one,

$$\begin{aligned} |\mathbb{E}[\widehat{p}_\sigma(a \mid x, z) \mid Z_{1:n}, X = x] - p_z(a \mid x)| &\leq \sum_{z'} \widehat{\omega}(z') |p_{z'}(a \mid x) - p_z(a \mid x)| \\ &\leq L(x) \sum_{z'} \widehat{\omega}(z') |z' - z|^{\gamma_b}. \end{aligned}$$

Thus the bias premise in Equation (107) follows with $C_b = L(x) C_K$ if the *empirical* weights satisfy $\sum_{z'} \widehat{\omega}(z') |z' - z|^{\gamma_b} \leq C_K \sigma^{\gamma_b}$. A moment bound for the population weights alone does not establish this finite-design hypothesis. For fixed bandwidth and positive limiting denominator, $n_{z'}/n \to p(z' \mid x)$ implies $\widehat{\omega}(z') \to \omega_\sigma(z' \mid x, z)$, so the conditional mixture converges to the population target.

Equation (108) gives a sufficient condition for beneficial smoothing: the bound on squared smoothing bias from Proposition G.3.2 is smaller than the variance reduction certified by Proposition G.3.3. For kernels maximized at the query delay, $n_{\text{eff}} \geq n_z$, with strict inequality whenever observations from other delays receive positive weight.

Under local homogeneity, the kernel and delay-conditioned variance expressions share the constant C_v used in Equation (107). More generally, let $C_{v,z} := p_z(a \mid x)\{1 - p_z(a \mid x)\}$ and $C_{v,\mathcal{N}} := \max_{z' \in \mathcal{N}} p_{z'}(a \mid x)\{1 - p_{z'}(a \mid x)\}$, where $\mathcal{N}$ contains the delays receiving positive kernel weight. The corresponding sufficient condition is

$$C_b^2 \sigma^{2\gamma_b} < \frac{C_{v,z}}{n_z} - \frac{C_{v,\mathcal{N}}}{n_{\text{eff}}}.$$

This reduces to Equation (108) when the constants coincide.

The conditional MSE analysis applies to a fixed or externally prescribed bandwidth. The population analysis characterizes the kernel target (Proposition G.1.1) and its interpolation endpoints and uniform bias bound (Remark G.2.1). The implemented bandwidth head is trained end to end using Equation (11), and its adaptation is evaluated empirically in Appendix B.4.

H Ground-Truth Policy Gradient

For the finite, enumerable environments used in our gradient- and credit-level analyses, we compute the required ground-truth quantities exactly by dynamic programming, following the procedure of Meulemans et al. (2023). We briefly summarize the construction here and refer to the original work for the full derivation.

For a fixed policy π_θ, we propagate the exact state-visitation probabilities through the known transition dynamics and compute the finite-horizon value functions by backward recursion,

$$Q_t^\pi(s,a) = \bar{r}_t(s,a) + \sum_{s'} p(s'|s,a) V_{t+1}^\pi(s'), \qquad V_t^\pi(s) = \sum_a \pi_\theta(a|s) Q_t^\pi(s,a),$$

with $V_T^\pi = 0$ and $\bar{r}_t(s,a) = \mathbb{E}[R_t \mid S_t = s, A_t = a]$. The exact policy gradient is then obtained by averaging the corresponding policy-gradient contributions over the exact state-visitation distribution,

$$g = \nabla_\theta J(\theta) = \sum_{t=0}^{T-1} \sum_s d_t^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s)\, Q_t^\pi(s,a).$$

Equivalently, $Q_t^\pi(s,a)$ may be replaced by the exact advantage $A_t^\pi(s,a) = Q_t^\pi(s,a) - V_t^\pi(s)$.

The same dynamic-programming enumeration provides the delay-conditioned (between source (s,t) and outcome) probabilities required by the contribution coefficients. For each (t,z,s,a,u) we therefore evaluate $p_{t,z}(u|s,a)$ and its policy-averaged counterpart exactly and obtain

$$w_{t,z}^\star(s,a,u) = \frac{p_{t,z}(u|s,a)}{\sum_b \pi_\theta(b|s) p_{t,z}(u|s,b)} - 1.$$

These exact gradients and contribution coefficients are used only as evaluation references; the learning algorithms themselves have access only to sampled on-policy trajectories. Definitions of the resulting bias, variance, SNR, and HCR metrics are given in Appendix I.4.

I Experimental Details and Hyperparameters

I.1 Baselines

In this study, we consider a broad range of policy-gradient estimators, which can be broadly categorized into temporal and causal (hindsight-based) approaches. REINFORCE is a return-to-go estimator that assigns each action credit for rewards observed later in the trajectory. When a discount factor $\gamma < 1$ is used, this credit decays with the temporal distance between the action and reward, reducing variance at the cost of introducing bias. In Table 2, we intentionally omit discounting and set $\gamma = 1$ to expose the underlying credit-assignment mechanism, treating discounting

Table 2: Comparison of policy-gradient estimators. Here, $Q(S_t, a)$ and $V(S_t)$ denote the state–action and state value functions, respectively, and $R_{\geq t} := \sum_{k\geq 0} R_{t+k}$ denotes the return-to-go from time t.

Method	**Policy gradient estimator** $\left(\widehat{\nabla}_\theta V^\pi(s_0)\right)$
REINFORCE	$\sum_{t\geq 0} \nabla_\theta \log \pi(A_t \mid S_t) \sum_{k\geq 0} R_{t+k}$
Advantage	$\sum_{t\geq 0} \nabla_\theta \log \pi(A_t \mid S_t) \left(\sum_{k\geq 0} R_{t+k} - V(S_t) \right)$
Q-critic	$\sum_{t\geq 0} \sum_{a\in\mathcal{A}} \nabla_\theta \pi(a \mid S_t) Q(S_t, a)$
TrajCV	$\sum_{t\geq 0} \nabla_\theta \log \pi(A_t \mid S_t) \left(R_{\geq t} - Q(S_t, A_t) - \sum_{t'>t} (Q(S_{t'}, A_{t'}) - V(S_{t'})) \right) + \ldots$ $\ldots \sum_{a\in\mathcal{A}} \nabla_\theta \pi(a \mid S_t) Q(S_t, a)$
COCOA	$\sum_{t\geq 0} \nabla_\theta \log \pi(A_t \mid S_t) R_t + \sum_{a\in\mathcal{A}} \nabla_\theta \pi(a \mid S_t) \sum_{k\geq 1} w(S_t, a, U_{t+k}) R_{t+k}$
HCA+	$\sum_{t\geq 0} \nabla_\theta \log \pi(A_t \mid S_t) R_t + \sum_{a\in\mathcal{A}} \nabla_\theta \pi(a \mid S_t) \sum_{k\geq 1} w(S_t, a, S_{t+k}) R_{t+k}$

primarily as a separate variance-reduction technique. Advantage further refines this temporal estimator by introducing a state-dependent baseline: the advantage $Q(S_t, A_t) - V(S_t)$ measures the value of the selected action relative to the policy-averaged value at the same state, thereby reducing variance without changing the expected policy gradient. Q-critic goes one step further by replacing the sampled-action Monte Carlo contribution with an expectation over actions under a learned state–action value function Q, with the critic itself typically learned through temporal-difference updates. TrajCV instead introduces trajectory-wise control variates constructed from learned value estimates, exploiting correlations across time to reduce the variance of Monte Carlo policy-gradient estimates while retaining trajectory-level information.

In contrast, COCOA and HCA+ perform causal, or hindsight-based, credit assignment. Rather than assigning credit primarily according to temporal proximity or ordering, they introduce a contribution coefficient w that estimates how strongly a past action contributed to a subsequently observed outcome. HCA+ uses the future state S as the hindsight target, whereas COCOA generalizes this construction to a reward-predictive outcome representation U, allowing credit to be assigned through a more general notion of rewarding outcome.

I.2 Environment Details

The environments are designed to separate distinct forms of temporal credit-assignment difficulty while keeping the underlying interaction structure simple. In the conveyor environments, the agent acts on the object at its current position and, except for the `Sticky Conveyor` described below, then progresses forward along the belt. Thus, navigation is not itself a learning problem; the relevant difficulty is determining which earlier actions are responsible for later rewards. Unless stated otherwise, the principal performance measure is successful collection of the task's target treasure.

Short/Long Key-door Conveyor. This environment contains a single key at the first cell, a door at cell $L-2$, and a potential treasure at the final cell $L-1$. Taking the key is the only consequential long-term decision and the door opens automatically when the agent reaches it with the key. Intermediate cells contain a fixed realization of left- and right-side apples with probability 0.8. Correctly collecting an apple produces a stochastic reward in $0.1, 0.9 \times 23/L$, whereas the final treasure yields $0.2 \times 23/L$. The experiments use short and long versions, principally $L = 23$ and $L = 43$, so increasing L directly increases the source–outcome delay without changing the underlying causal structure. The task therefore measures whether an early enabling action can be credited for a distant outcome in the presence of many intervening stochastic rewards. Its closest behavioral analogue is

the classical delayed-reinforcement paradigm, in which an operant response and its reinforcer are separated in time (Lattal, 2010). This task is studied in the COCOA paper (Meulemans et al., 2023).

Ordered-keys Conveyor. The Ordered-keys environment (`Ordered-keys Conveyor`) places two distinct types of keys, A and B, at separate randomly sampled cells within positions 1–9. The agent must collect *both* keys before opening the door at $L-2$. Multiple type-A and type-B keys appear on the conveyor, and the agent can carry at most two keys at a time; the currently held keys are visible. To satisfy the prerequisite, the agent must collect the keys in the correct order: the first key in the packet must be type A and the second type B. The packet follows a FIFO convention. Other cells before the door contain stochastic apple rewards and traps. If the agent selects the key-pick action on a trap, both previously collected keys are erased, requiring the agent to reacquire them in the correct order. The terminal treasure is available only if both keys remain intact and correctly ordered when the door is opened. The task isolates conjunctive credit assignment: neither key is sufficient by itself, but both earlier decisions jointly enable the delayed reward. This has a useful structural analogy to positive-patterning experiments in associative learning, $A-$, $B-$, $AB+$, in which individual cues are nonreinforced but their conjunction predicts reinforcement (Lachnit & Lober, 2001). The present task differs by placing the two enabling causes at different points in a sequential trajectory.

Colored-keys Conveyor. The Colored-keys environment samples a target color, red, blue, or green, at the beginning of each episode and exposes this target as a persistent contextual cue. One key of each color is placed at a distinct randomly sampled position in the early key region (cells 1–9 for $L = 23$), while the remaining cells before the door contain stochastic left- and right-side apple rewards. The agent can hold only one key at a time, and picking up a new key replaces the previously held key. At cell $L-2$, the door opens only if the agent selects the open-door action while carrying the key whose color matches the episode's target; the corresponding treasure reward is then obtained at the final cell $L-1$. Thus, the relevant long-term value of a key-pick action is determined jointly by the encountered key and the episode-level context. The matching key is an enabling cause of the delayed treasure, whereas non-target keys provide no such credit and can overwrite a previously correct choice. The task therefore isolates context-dependent delayed credit assignment, requiring the agent to associate a distant outcome with the appropriate early decision despite many intervening stochastic rewards. Structurally, it is analogous to a conditional-discrimination problem, in which an explicit contextual cue determines which otherwise similar choice is rewarded.

Multicolored-keys Conveyor. The Multicolored-keys environment uses a global preference ordering red $<$ green $<$ yellow $<$ blue $<$ orange. At the beginning of each episode, three colors are sampled without replacement. Their keys are placed in the early key region (cells 1–5 for $L = 23$), and the corresponding doors are placed in the final seven cells. The agent can hold one key at a time. Opening the matching door for the least-, middle-, or highest-ranked sampled color yields respectively 0.25, 0.5, or 1.0 times the base treasure reward, after which subsequent rewards are suppressed. The reported success metric counts only the collection of the highest-ranked treasure available in each episode. Thus, credit depends not simply on a key's absolute identity, but on its rank relative to the subset of colors present in that episode under a global ranking that remains fixed throughout training. This relative ranking therefore acts as an implicit form of context. A related cognitive paradigm is transitive-inference learning, in which subjects learn or exploit an ordered relation among stimuli and use that ordering to guide choices (Bryant & Trabasso, 1971). The analogy is structural rather than exact: here the challenge is selecting the maximum-ranked member of a sampled subset rather than inferring a novel pairwise relation.

XOR-keys Conveyor. The XOR-keys environment places keys A and B at two distinct positions in cells 1–8. The door at $L-2$ opens only when the agent has collected exactly one of the two keys: $A \oplus B$. Collecting neither key and collecting both keys therefore lead to failure, whereas either single-key trajectory enables the treasure. The value of taking one key consequently depends on whether the other key has already been taken, making credit intrinsically non-additive over the action history. The closest standard associative-learning analogue is negative patterning, $A+$, $B+$, $AB-$, in which either component alone predicts reinforcement but their compound does not (Lachnit & Lober, 2001). XOR adds the additional nonreinforced "neither" case and embeds the relation sequentially.

Poisoned-apples Conveyor. The Poisoned-apples environment contains safe and poisonous apples before a door at $L - 2$. At the canonical length $L = 23$, each intermediate cell is safe with probability 0.8, poisonous with probability 0.1, and empty otherwise. A collected safe apple produces reward 0.1 or 0.9 with equal probability. The first collected poison apple instead produces an immediately attractive reward of 0.9 or 1.1 and sets a latent poisoned flag; later poison apples yield zero. Safe apples remain rewardable after poisoning, but the final treasure of 1.05 is available only if the agent opens the door and has never collected a poison apple. The task therefore separates immediate reward from delayed consequence: an action can appear locally favorable while eliminating a later outcome. A classical psychological parallel is conditioned taste-aversion learning, in which an ingestive cue becomes associated with a delayed adverse consequence (Garcia & Koelling, 1966); the present environment turns that delayed adverse consequence into a sequential decision and credit-assignment problem.

Coupon-or-vault. The Coupon-or-vault environment is designed to produce an early finite-data failure of a learned hindsight model. At the initial state, action 0 chooses a Coupon branch and action 1 chooses a Vault branch. The Coupon branch produces a deterministic reward of 1 at the next position. The Vault branch instead contains eight fully observed binary gates with cues $(0, 1, 0, 1, 1, 0, 1, 0)$. At every gate the agent must choose the action matching the displayed cue; failing any gate permanently makes the trajectory ineligible for the Vault. Both branches subsequently traverse eight action-independent nuisance-reward cells, each producing -16 or $+16$ with equal probability. A successful Vault trajectory finally yields reward 384. Under an initially uniform gate policy, $Q(\text{Vault}) = 384(1/2)^8 = 1.5 > Q(\text{Coupon}) = 1$, but including the initial branch choice, a successful Vault outcome occurs in only $1/512$ of initial-policy episodes. Thus the globally superior decision is supported by a rare delayed outcome, whereas the inferior alternative produces frequent immediate evidence. This is closely related to the "decisions from experience" literature, where rare outcomes may be underrepresented or underweighted when behavior is learned from limited experience (Hertwig et al., 2004).

Delay Dilution. Delay Dilution isolates loss of temporal specificity caused by pooling outcomes across delays. In the experiments $L = 17$. Only the action at position 0 is consequential: action 0 selects the baseline branch and action 1 the informative branch. Positions $1, \ldots, L - 2$ are nuisance stations that independently emit the same scalar reward $R = 1$ with probability 0.5, regardless of the initial action. At the final station, the same reward value occurs with probability 0.1 after the baseline action and 0.9 after the informative action. All actions after the initial choice are inert. Hence the reward identity alone does not distinguish the many action-independent occurrences from the one action-informative occurrence; only source–outcome delay separates them. Pooling across delays therefore attenuates the informative contribution, whereas delay conditioning isolates it. This manipulation is related to experimental work showing that temporal order and temporal predictability are informative cues in human causal learning (Lagnado & Sloman, 2006; Greville & Buehner, 2010).

Multi-Length Conveyor. The Multi-Length Conveyor is a diagnostic environment for the delay-conditioned model and its learned smoothing bandwidth. The first action selects one of several conveyors; all subsequent actions are inert. Every conveyor terminates in the same deterministic treasure, but a conveyor of length L_i delivers it at delay $z = L_i - 1$. The bandwidth experiments in Appendix B.4 use custom collections of nearby and separated delays. Because every initial choice obtains the same eventual return, the ordinary delay-aggregated advantage is zero. Nevertheless, the delay-resolved contribution is highly informative: each choice shifts the reward probability mass to its own specific delay. The task therefore tests whether a method preserves the temporal location of otherwise identical outcomes. Its closest experimental-psychology analogue is interval-timing and temporal-bisection work, in which subjects discriminate or generalize across different temporal intervals (Church & Deluty, 1977); here, however, temporal information is used for causal credit rather than explicit duration classification.

Sticky Conveyor. The Sticky Conveyor starts from the same causal structure as the canonical conveyor: a key is located at cell 0, apples occupy a fixed subset of intermediate cells, and possession of the key determines whether treasure appears at the final cell. Its defining manipulation is that a successful pickup consumes a timestep without advancing the belt. After taking a key or apple,

the object disappears but the agent remains at that cell for one additional decision; other actions advance normally. Consequently, if N_{pick} objects are collected, the key-to-treasure delay is $z = (L-1) + N_{\text{pick}}$, where L is the conveyor length. The same causal relation is therefore observed over a policy-dependent distribution of delays rather than at one fixed delay. This task tests whether the learned kernel shares information across nearby delays when data are sparse and becomes more local as the relevant delay structure is resolved. The manipulation is closely related conceptually to findings that causal learning is affected by the consistency and predictability of cause–effect timing (Greville & Buehner, 2010).

Stochastic-interleaving. The Interleaving environment consists of several independent contexts. In a query state for context c, two objects are displayed, one drawn from that context's rewarded set and one from its unrewarded set, with their left–right order randomized. The agent selects left or right, and whether this selection was correct is stored as a context-specific binary key. When that context later appears in an answer state, the current action is irrelevant and a context-dependent reward is delivered if and only if the earlier query choice was correct. The generic environment can leave several contexts unresolved simultaneously, so actions from unrelated contexts may intervene between a query and its corresponding outcome. The task is structurally related to delayed matching-to-sample and other delayed conditional-discrimination paradigms, in which information presented at one time determines the correct response after a retention interval (Adamson et al., 2000). Unlike those paradigms, the intended difficulty here is assigning the later outcome to the relevant earlier decision rather than measuring memory capacity itself. This task is also studied in COCOA (Meulemans et al., 2023).

I.3 Atari: Montezuma's Revenge

We evaluate HDCA on Montezuma's Revenge as a more complex long-horizon credit-assignment problem. Following the same principle as in our synthetic experiments, we aim to reduce the confounding effects of representation learning and initial exploration. We therefore use the Atari environment with a symbolic state representation rather than raw pixels. The agent operates with the full Atari action space, with four emulator frames per decision, and trajectories are truncated to 256 agent decisions.

State representation. The policy receives a single 84-dimensional symbolic state vector. It summarizes the current room and game status, the agent's position and motion, inventory, relevant collectibles and enemies, and dynamic room variables such as doors and platforms. We additionally retain a small number of temporal features, such as object velocities, when the instantaneous object configuration alone is insufficient to characterize the dynamics. Static room geometry is omitted because it is determined by the room identity. Thus, the representation preserves the sequential control and delayed-credit structure of the game while largely removing the difficulty of learning visual features.

Demonstrations and behavior-cloning pretraining. Pure on-policy learning from a randomly initialized policy rarely reaches the sparse rewards required to expose the relevant credit-assignment problem. We therefore collect only a small number of successful demonstrations covering the sequence of reaching the key and subsequently opening the door. The same demonstrations are shared across methods and are used to pretrain each policy by behavior cloning (BC) before RL fine-tuning. We deliberately keep this demonstration set small: the resulting BC policies achieve only approximately 20% task success at the beginning of RL training. Hence, BC does not solve the task, but provides sufficient initial coverage for all methods to encounter successful trajectories and learn from delayed rewards. Demonstrations are used for initialization only and are not mixed into subsequent on-policy RL data.

Auxiliary pretraining and warm-up. Learned auxiliary components are initialized consistently with the pretrained policy. Hindsight-based methods are first trained on the demonstration data using their corresponding contribution-learning objectives. As described in the main text, we additionally use a short warm-up period for components that remain untrained or only partially pretrained, including value or action-value functions and hindsight models. During this period the BC policy is held fixed while these modules are trained from several thousand interaction samples. Policy fine-tuning

begins only afterwards. This avoids pairing an already behavior-cloned policy with randomly initialized credit or value estimators and allows the subsequent comparison to primarily reflect differences in credit assignment. After this initialization stage, all methods learn from on-policy interaction using the original environment reward.

I.4 METRICS

We evaluate the quality of the learning signal at two complementary levels. Following Meulemans et al. (2023), we use normalized bias, variance, and signal-to-noise ratio (SNR) to measure the quality of the policy-gradient estimator. We additionally use the Hindsight Credit Reconstruction error (HCR) to measure whether individual delayed rewards are attributed to the correct earlier decisions at the correct delays. Ground-truth quantities are computed exactly by the dynamic-programming procedure described in Appendix H.

Policy-gradient bias, variance, and SNR. Let

$$g := \nabla_\theta J(\theta)$$

denote the exact policy gradient, let $\widehat{g}(\tau)$ denote the gradient estimate produced from an on-policy trajectory τ, and let

$$\bar{g} := \mathbb{E}_{\tau\sim\pi_\theta}[\widehat{g}(\tau)]$$

denote its expected value. We use the normalized squared bias and variance

$$\mathrm{Bias} = \frac{\|\bar{g} - g\|_2^2}{\|g\|_2^2}, \qquad \mathrm{Var} = \frac{\mathbb{E}_{\tau\sim\pi_\theta}\left[\|\widehat{g}(\tau) - \bar{g}\|_2^2\right]}{\|g\|_2^2}.$$

Bias therefore measures systematic displacement of the expected update from the true policy gradient, whereas variance measures trajectory-to-trajectory fluctuation around that expected update. Following Meulemans et al. (2023), we define the policy-gradient SNR as

$$\mathrm{SNR} = \frac{\|g\|_2^2}{\mathbb{E}_{\tau\sim\pi_\theta}\left[\|\widehat{g}(\tau) - g\|_2^2\right]}.$$

At the population level, $\mathrm{SNR}^{-1} = \mathrm{Bias} + \mathrm{Var}$. Thus, lower bias and variance and higher SNR correspond to a more accurate policy-gradient signal. In our exact-MDP experiments, g and $\bar{g}$ are computed by dynamic programming, while the remaining trajectory expectation is estimated from the evaluation trajectories.

Delay-resolved credit reconstruction. The preceding quantities assess the final gradient but do not reveal whether a method assigned a delayed reward to the correct source action and delay. For a source state s, candidate action a, and delay $z \geq 1$, we define the ground-truth delay-resolved credit as

$$C_z^\star(s,a) := \mathbb{E}[R_{t+z} \mid S_t = s, A_t = a] - \mathbb{E}[R_{t+z} \mid S_t = s].$$

This quantity is computed exactly from the MDP and measures the action-dependent part of the reward occurring specifically at delay z. Summing over delays recovers the future component of the exact advantage.

Let $\widehat{C}_z(s,a)$ denote the corresponding expected credit assigned by a method. For a contribution-coefficient estimator this is

$$\widehat{C}_z(s,a) = \mathbb{E}[\widehat{w}_z(s,a,U_{t+z})R_{t+z} \mid S_t = s]\,,$$

where the contribution coefficient is taken in its centered form $\widehat{w}_z = \widehat{h}_z/\pi_\theta - 1$. For HDCA, $\widehat{C}_z$ denotes the credit after applying the same causal–temporal mixing mechanism as in the policy-gradient estimator.

Let

$$d^\pi(s) := \sum_{t=0}^{T-1} \Pr_\pi(S_t = s), \qquad \mu^\pi(s,a) := d^\pi(s)\pi_\theta(a \mid s),$$

Table 3: Task-specific critical states used for evaluating HCR. Critical states are the decision points at which the agent's action can affect the task-defining delayed outcome. States within the same row are grouped when they are equivalent for the underlying credit-assignment problem.

Task	Critical states
Short/Long Key-door Conveyor	The key state and the door state. These are the two states directly involved in enabling and obtaining the delayed treasure; intermediate apple states are not treated as critical.
Ordered-keys Conveyor	The states containing type-A and type-B keys, treated as two separate critical-state groups. These are the decisions that determine whether the required key configuration is available at the door.
Colored-keys Conveyor	All key states, divided into target-key and non-target-key states according to whether the encountered key matches the current target color. Non-target colors are grouped together because their individual color identities are irrelevant to the task-defining decision.
Multicolored-keys Conveyor	All key states, grouped according to the key's preference rank among the colors sampled in the episode: preferred, second-preferred, and least-preferred. The grouping is by reward-relevant rank rather than absolute color identity.
XOR-keys Conveyor	The states containing key A and key B, treated separately. These are the two decisions whose joint pickup history determines whether the exclusive-or condition for obtaining the treasure is satisfied.
Delay Dilution	The initial-choice state at the beginning of the episode. This is the single consequential decision: it changes the probability of the informative delayed reward, while all subsequent actions are irrelevant to that outcome.
Coupon-or-vault	The initial branch-choice state and each vault-gate state. Individual gates are kept as separate critical-state groups because later gates are reached only after the preceding gates succeed and therefore have different visitation probabilities and credit structure.

and let $\mathcal{S}_{\mathrm{crit}}$ denote the set of critical source states for the task. We define

$$\mathrm{HCR} = \frac{\sum_{s\in\mathcal{S}_{\mathrm{crit}}}\sum_{a\in\mathcal{A}}\mu^{\pi}(s,a)\sum_{z}\big(\widehat{C}_z(s,a)-C^{\star}_z(s,a)\big)^2}{\sum_{s\in\mathcal{S}_{\mathrm{crit}}}\sum_{a\in\mathcal{A}}\mu^{\pi}(s,a)\sum_{z}\big(C^{\star}_z(s,a)\big)^2}.$$

HCR therefore measures error before summing credit over delays: assigning the right total amount of credit but placing it at the wrong delay is still penalized. It is normalized by the ground-truth credit energy, so $\mathrm{HCR}=0$ corresponds to exact delay-resolved credit assignment, whereas assigning zero credit gives $\mathrm{HCR}=1$. HCR is computed from expected assigned credit under the exact MDP and therefore measures systematic credit-assignment error rather than sampling variance; the latter is captured separately by the policy-gradient variance above.

For figures shown on a decibel scale, we report $10\log_{10}$ of the corresponding dimensionless ratio. Thus, higher values are preferable for SNR, whereas lower values are preferable for bias, variance, and HCR. The task-specific sets $\mathcal{S}_{\mathrm{crit}}$ used for HCR are specified in Table 3. These states are intuitively more decisive than other states for obtaining rewards, in particular the treasure rewards. In our analysis, we may further divide the critical states into different groups. However, in the reported figures, we average across all critical states to keep the observations clear and straightforward.

I.5 Hyperparameters

In the following, we provide a concise summary of the hyperparameters used for each method. For the baseline methods, we used the hyperparameters from the original COCOA implementation[3]. Since the tasks considered in this work are relatively similar in terms of state and action spaces and environment dynamics, we directly transfer the hyperparameters to the new tasks, except for the policy learning rate. We also conduct a hyperparameter sensitivity analysis across these tasks. For

[3]https://github.com/seijin-kobayashi/cocoa

Table 4: Shared training hyperparameters for the main conveyor experiments. Unless otherwise noted, the same settings are used for the canonical Key-door Conveyor and the scaled conveyor tasks (Ordered-keys, Colored-keys, Multicolored-keys, XOR-keys, and Poisoned-apples). L denotes the episode horizon.

Hyperparameter	**Value**
Policy network	MLP $(64, 64)$
Hidden-layer activation	ReLU
Policy optimizer	AdamW
Policy-optimizer weight decay	3×10^{-4}
Trajectories collected per policy update	8
Evaluation trajectories per checkpoint	512
Action exploration, ϵ	0.05
Return discounting	undiscounted ($\gamma = 1$)
Entropy-bonus coefficient	Canonical: 0.03, scaled with conveyor length

Table 5: Hyperparameters of the hindsight-based baselines in the main conveyor experiments. All methods use a HyperNet hindsight predictor with a 256-unit hidden representation, similar to Meulemans et al. (2023). All hindsight models are optimized with AdamW and 1×10^{-4} weight decay.

Method	**Policy LR**	**Outcome representation**	**Hindsight LR**	**Zero-reward outcomes**
HCA+	3×10^{-4}	Future state	1×10^{-3}	Included
COCOA-reward	3×10^{-4}	Scalar reward	3×10^{-3}	Excluded
COCOA-feature	3×10^{-4}	Learned reward-predictive feature	3×10^{-3}	Excluded

our method, hyperparameters that were already present in the baseline implementations (COCOA-reward and Advantage) in the original repository were employed using their default values.

We also performed a hyperparameter search over the policy learning rate for all methods, considering $\mathrm{lr} \in \{1 \times 10^{-3}, 3 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-4}\}$. The search was conducted on `Multicolored-keys Conveyor` and `Colored-keys Conveyor`, with performance averaged over three random seeds. The selected learning rate was the same as the default value provided in the baseline repository.

In addition, following Patterson et al. (2024), who suggest several approaches to ensure fair comparisons between methods in terms of hyperparameters, we conducted a hyperparameter sensitivity analysis (Figure 16). Specifically, we evaluated our method and the baselines on four tasks using 12 different hyperparameter configurations per method, varying the learning rate, exploration budget, and network size. Each configuration was evaluated over three random seeds.

To provide a more fine-grained comparison throughout training, we divided the training budget into four equal intervals and reported a violin plot of the percentage of collected treasure rewards for each interval. We used half of the original training budget for the sensitivity analysis. The results show that HDCA learns faster, with performance improvements emerging from the early stages of training, and exhibits lower variability, particularly in `XOR-keys` and `Multicolored-keys`. Moreover, as shown in Figure 16, HDCA achieves higher mean final performance than the baseline methods across different configurations. Together, these results suggest that HDCA is more robust to hyperparameter choices than the baselines.

J Algorithm Pseudocode

Algorithm 1 summarizes HDCA. Each iteration has three stages: (i) the hindsight ensemble supplies a contribution coefficient and a reliability mixing coefficient for every source–outcome pair; (ii) these are combined into the hybrid policy gradient; and (iii) the ensemble and bandwidth networks are then updated on the kernel-smoothed loss (Equation (11)).

Table 6: Additional hyperparameters for COCOA-feature. The reward-predictive feature extractor is trained once after the initial burn-in period and is subsequently held fixed.

Hyperparameter	**Key-door Conveyor**	**Other Conveyors**	**Interleaving**
Feature dimension	128	128	Observation dimension
Reward-prediction readout	Action-specific	Shared across actions	Shared across actions
Feature optimizer	AdamW	AdamW	AdamW
Feature learning rate	3×10^{-3}	3×10^{-3}	1×10^{-3}
Feature-training steps	10,000	20,000	30,000
Feature-training batch size	$64L$	$128L$	720
Feature replay capacity	$64L$	$128L$	720
Feature-extractor weight decay	0.001	0.001	0.003

Table 7: Hyperparameters of HDCA in the main conveyor experiments. The causal weight $\widehat{\alpha}$ controls the relative contributions of the learned causal and temporal estimators. The hyperparameters have the same values as the COCOA-reward and Advantage baselines.

Hyperparameter	**Value**
Policy and causal estimator	
Policy learning rate	3×10^{-4}
Hindsight ensemble size	5
Hindsight model	MLP, 256 hidden units
Hindsight optimizer	AdamW
Hindsight learning rate	3×10^{-3}
Hindsight updates per policy update	1
Zero-reward outcomes	Excluded
Policy prior in hindsight posterior	Yes
Temporal estimator	
Temporal estimator	REINFORCE
Value network	MLP, 256 hidden units
Value learning rate	1×10^{-3}
Value-target trace parameter	$\lambda = 1$
Value updates per policy update	1
Delay conditioning	
Delay representation	16-dimensional Fourier features
Maximum represented delay	Episode horizon
Kernel smoothing	
Kernel	Gaussian
Bandwidth predictor	MLP, 64 hidden units
Smoothing range	± 4 delay steps

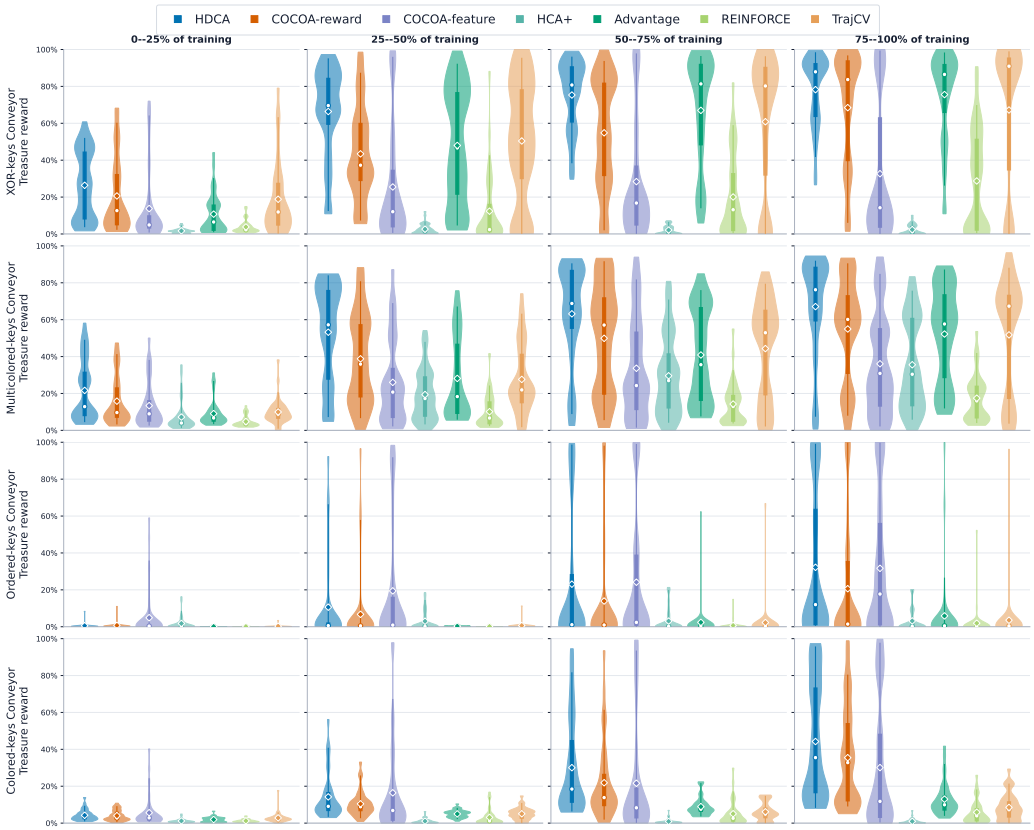


Figure 16: **Hyperparameter sensitivity across training.** Violin plots show the distribution of treasure-reward collection for HDCA and the baselines on the `XOR-keys`, `Multicolored-keys`, `Ordered-keys`, and `Colored-keys` Conveyor tasks over four equal intervals of training. Each method is evaluated under 12 hyperparameter configurations, varying the learning rate, exploration budget, and network size, with three random seeds per configuration. The distributions reveal both learning progress and sensitivity to hyperparameter choice throughout training; these sensitivity experiments use half of the training budget of the main experiments.

Algorithm 1 Hybrid Delay-aware Credit Assignment (HDCA)

Require: policy π_θ; ensemble size M; number of iterations N

Initialize $\{h_{\phi_m}, \sigma_{\psi_m}\}_{m=1}^{M}$ independently and store ϕ_m^0
for $n = 1, \dots, N$ **do**
  Sample a batch $\mathcal{B}$ of trajectories with π_θ
  Per-pair credit
  for each trajectory in $\mathcal{B}$ and each pair $(t, z) \in \mathcal{I}_T$ **do**
    $x_{t,z} \leftarrow (t, S_t, U_{t+z}, z), \quad \bar{h}_\phi \leftarrow \frac{1}{M} \sum_m h_{\phi_m}(\cdot \mid x_{t,z})$
    $\widehat{w}_{t,z} \leftarrow \bar{h}_\phi / \pi_\theta(\cdot \mid S_t) - 1$ ▷ Equation (4)
    $\widehat{V}_\varepsilon^{(t,z)} \leftarrow 1 - \left\|\bar{h}_\phi\right\|_2^2, \quad \widehat{V}_\Delta^{(t,z)} \leftarrow \frac{M}{M-1} \mathcal{U}^2(x_{t,z})$ ▷ Equation (8)
    $\widehat{\alpha}_{t,z} \leftarrow \widehat{V}_\varepsilon^{(t,z)} / \big(\widehat{V}_\varepsilon^{(t,z)} + \widehat{V}_\Delta^{(t,z)}\big)$
  end for
  Policy update
  $\widehat{G}_H \leftarrow \Pi + \sum_{(t,z)} \left[\widehat{\alpha}_{t,z}\, \widehat{G}_{CD}^{(t,z)} + (1 - \widehat{\alpha}_{t,z})\, G_R^{(t,z)}\right]$ ▷ Equations (5) and (6)
  $\theta \leftarrow \theta + \eta\, \widehat{G}_H$
  Hindsight update
  for $m = 1, \dots, M$ **do**
    For each query (x, z) in $\mathcal{B}$, set $\sigma \leftarrow \sigma_{\psi_m}(x, z)$ and weights $\omega_\sigma(z' \mid x, z)$
    $\mathcal{L}_m \leftarrow \mathrm{mean}_{(x,z)}\ \widetilde{\mathcal{L}}_\sigma(h_{\phi_m}; x, z) + \lambda_{\mathrm{reg}} \left\|\phi_m - \phi_m^0\right\|_2^2$ ▷ Equation (11)
    if n is odd **then**
      Update ϕ_m on $\mathcal{L}_m$ with ψ_m fixed
    else
      Update ψ_m on $\mathcal{L}_m$ with ϕ_m fixed
    end if
  end for
end for