Improving Reward-Based Hindsight Credit Assignment

Aditya A. Ramesh, Jiamin He, Jürgen Schmidhuber, Martha White

Keywords: Credit Assignment, Hindsight, Sample-efficiency

Summary

Accurately attributing credit to past actions is crucial for sample-efficient reinforcement learning. While temporal-difference learning with λ-returns is the most commonly used approach, it assigns credit based on the temporal proximity of actions and outcomes—a heuristic that may be overly simple in complex environments. Hindsight-based approaches offer an alternative by explicitly crediting actions that were critical to achieving specific outcomes. Recent work has shown that conditioning hindsight predictions on future rewards, rather than states, can improve credit assignment. However, we show that the associated algorithm, Counterfactual Contribution Analysis (COCOA), suboptimally handles immediate rewards in noisy settings, degenerating into a high-variance estimator even with perfect hindsight. We introduce Reward Hindsight Credit Assignment (R-HCA), which extends hindsight reweighting to incorporate immediate rewards. We prove that the variance of our R-HCA is no greater than that of COCOA in contextual bandit problems and sparse-reward Markov Decision Processes. Our experiments demonstrate that R-HCA with learned hindsight models outperforms COCOA in domains that require long-term credit assignment with noisy intermediate rewards. We also characterize the conditions under which reward-based hindsight is preferable to standard actor-critic methods.

Contribution(s)

1. We identify a failure mode of COCOA (Meulemans et al., 2023), a reward-based hindsight credit assignment algorithm. We show that it degenerates into a high-variance estimator due to how it handles immediate rewards in noisy environments.
   Context: Hindsight-based approaches offer a promising alternative to temporal-difference learning by explicitly linking past actions to future outcomes rather than relying on temporal proximity. Recently, Counterfactual Contribution Analysis (COCOA) (Meulemans et al., 2023) was introduced to use future rewards for hindsight-based credit assignment.
2. We introduce a modified algorithm, Reward Hindsight Credit Assignment (R-HCA), that incorporates immediate rewards through hindsight reweighting. We empirically validate R-HCA with learned hindsight models, demonstrating improved sample efficiency over COCOA in domains requiring long-term credit assignment with noisy intermediate rewards.
   Context: Our main experiments allow reward noise and the delay between consequential actions and outcomes to be varied independently, enabling a controlled comparison across a spectrum of credit assignment difficulty. We also characterize the conditions under which reward-based hindsight credit assignment is most beneficial, showing that traditional actor-critic baselines remain preferable when informative feedback closely follows actions.
3. We prove that the variance of R-HCA is no greater than that of COCOA in contextual bandit problems and sparse-reward Markov Decision Processes (MDPs). For general MDPs, we provide a variance decomposition. We show that while R-HCA reduces the variance associated with the immediate reward term, the overall comparison depends on the covariance between immediate reward differences and future returns, indicating that there are specific MDP structures in which COCOA yields lower variance.
   Context: Our theoretical analysis of hindsight-based credit assignment assumes access to a perfect hindsight model to isolate the mathematical properties of the estimators.

Improving Reward-Based Hindsight Credit Assignment

Aditya A. Ramesh[1], **Jiamin He**[2], **Jürgen Schmidhuber**[3,1], **Martha White**[2]
{aditya, juergen}@idsia.ch, {jiamin12, whitem}@ualberta.ca

[1]**The Swiss AI Lab, IDSIA USI-SUPSI**
[2]**University of Alberta and the Alberta Machine Intelligence Institute**
[3]**Center of Excellence for Generative AI, KAUST**

Abstract

Accurately attributing credit to past actions is crucial for sample-efficient reinforcement learning. While temporal-difference learning with λ-returns is the most commonly used approach, it assigns credit based on the temporal proximity of actions and outcomes—a heuristic that may be overly simple in complex environments. Hindsight-based approaches offer an alternative by using a model that leverages future information to more explicitly credit previous actions that were critical to achieving specific outcomes. Recent work has shown that conditioning hindsight predictions on future rewards, rather than states, can improve credit assignment. However, we show that the associated algorithm, Counterfactual Contribution Analysis (COCOA), suboptimally handles immediate rewards in noisy settings, degenerating into a high-variance estimator even with perfect hindsight. We introduce Reward Hindsight Credit Assignment (R-HCA), which extends hindsight reweighting to immediate rewards. We prove that the variance of R-HCA is no greater than that of COCOA in contextual bandit problems and sparse-reward Markov Decision Processes. Our experiments demonstrate that R-HCA with learned hindsight models outperforms COCOA in domains that require long-term credit assignment with noisy intermediate rewards. We also characterize the conditions under which reward-based hindsight is preferable to standard actor-critic methods.

1 Introduction

Building on the principles of dynamic programming (Bellman, 1954), temporal-difference (TD) learning has become a central component of modern reinforcement learning (RL). The λ-return (Watkins, 1989) and its online counterpart, TD(λ) (Sutton, 1988), form the basis for credit assignment in many widely adopted RL algorithms.

Although these methods are simple and reliable, they rely on the temporal proximity of actions and outcomes for credit assignment (Daley et al., 2024), which can hinder their ability to efficiently solve certain challenging credit assignment problems (Pignatelli et al., 2024). As alternatives to λ-return-based credit assignment, hindsight-based approaches have been proposed, including Upside-Down RL (Schmidhuber, 2019) and Hindsight Credit Assignment (HCA; Harutyunyan et al., 2019).

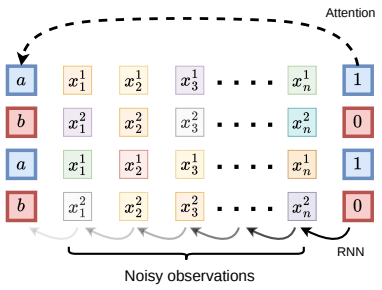


Figure 1: Just as attention mechanisms bypass noisy intermediate tokens (x_i^j) to directly associate critical inputs (a/b) with outcomes (1/0), hindsight-based approaches can skip over noisy intermediate steps to assign credit to critical past actions, as demonstrated in the Umbrella problem in Section 3.

In HCA, the agent learns a hindsight model that predicts a distribution over actions in a particular state using future outcomes (e.g., returns, states, or rewards) as additional information. The prediction from the hindsight model is used to re-weight the relevance of past actions toward achieving a particular outcome. This enables the agent to explicitly associate consequences across distant transitions without incurring an exponential decay like with λ-returns (Arjona-Medina et al., 2019). Therefore, HCA may present advantages over λ-return-based temporal difference, similar to some benefits that attention-based architectures (e.g., Schmidhuber (1992); Bahdanau et al. (2015); Vaswani et al. (2017)) provide over standard recurrent neural networks (RNNs) for sequence modeling (see Figure 1). Additionally, the hindsight re-weighting is computed for all actions, not just the one taken, allowing HCA to assign credit to counterfactual actions.

Previous work by Meulemans et al. (2023) shows that using future states for hindsight reweighting can cause HCA to degenerate into the high-variance REINFORCE (Aleksandrov et al., 1968; Williams, 1987) approximation of stochastic gradient descent (Robbins & Monro, 1951; Amari, 1967). To address this, they propose a policy gradient estimator, Counterfactual Contribution Analysis (COCOA), that uses future rewards for hindsight predictions. They show that conditioning on rewards rather than states filters out task-irrelevant information about prior actions, thereby reducing the variance of the hindsight-based policy gradient estimator.

In this paper, we show that the variance of the COCOA policy gradient estimator can be excessively high even with perfect hindsight predictions. This higher variance arises because COCOA handles immediate rewards through a REINFORCE-style update while using hindsight reweighting only for future rewards. We propose Reward Hindsight Credit Assignment (R-HCA), a modified algorithm that resolves this problem by incorporating immediate rewards through hindsight reweighting. Assuming access to a perfect hindsight model, we prove that R-HCA has variance no greater than COCOA in contextual bandits and sparse-reward MDPs. For general MDPs, we provide a variance decomposition showing that the comparison depends on the interaction between immediate and future reward terms, and we identify MDP structures where COCOA can have lower variance. Empirically, we demonstrate that R-HCA with learned hindsight models consistently outperforms COCOA in domains that require long-term credit assignment with noisy intermediate rewards. We also characterize the settings where reward-based hindsight credit assignment is most beneficial, showing that standard actor-critic baselines remain preferable when informative feedback closely follows actions.

2 Preliminaries

Consider a Markov Decision Process (MDP) with state space $\mathcal{S}$, action space $\mathcal{A}$, reward space $\mathcal{R}$, and discount factor γ. At each time step t, the agent observes state S_t, takes action $A_t \sim \pi(\cdot|S_t)$, and receives reward R_t, generating a trajectory $\tau = (S_0, A_0, R_0, S_1, A_1, R_1, \dots)$. We denote the distribution over trajectories starting from state s (or state-action pair (s, a)) under policy π by $\mathcal{T}(s, \pi)$ (or $\mathcal{T}(s, a, \pi)$). The discounted return is $Z(\tau) = \sum_{t\geq 0} \gamma^t R_t$, the state value is $V_\pi(s) = \mathbb{E}_{\tau\sim\mathcal{T}(s,\pi)}[Z(\tau)]$, the action value is $Q_\pi(s, a) = \mathbb{E}_{\tau\sim\mathcal{T}(s,a,\pi)}[Z(\tau)]$, and the advantage is $A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s)$.

2.1 Advantage Actor-Critic

In the actor-critic framework (Konda & Tsitsiklis, 1999; Sutton et al., 1999), the policy (actor) is updated via a discounted approximation to the policy gradient using generalized advantage estimates (Schulman et al., 2015),

$$\hat{\nabla} V_\pi(S_0) = \sum_{t\geq 0} \nabla_\theta \log \pi_\theta(A_t|S_t) \hat{A}_t^{GAE(\gamma,\lambda)}. \tag{1}$$

The advantages are estimated using a learned critic $V : \mathcal{S} \to \mathbb{R}$. Let $\delta_t = R_t + \gamma V(S_{t+1}) - V(S_t)$, then $\hat{A}_t^{GAE(\gamma,\lambda)} = \sum_{l\geq 0} (\gamma\lambda)^l \delta_{t+l}$. The critic is trained with temporal-difference λ-return

targets (Sutton et al., 1998). This setup is standard in on-policy algorithms such as A2C (Mnih et al., 2016) and PPO (Schulman et al., 2017).

2.2 Hindsight Credit Assignment

Hindsight Credit Assignment (HCA) uses a learned hindsight distribution to estimate targets for policy updates. Here we describe a generalization of the original State-HCA (Harutyunyan et al., 2019) with rewarding outcomes, as presented by Meulemans et al. (2023).

HCA approaches rely on a hindsight distribution $h_k^\pi(a|s,u)$, which encodes the probability $P(A_t = a|S_t = s, U_{t+k} = u)$ that the agent chose action a in state s given that it observes a future outcome u, k steps after observing s, while following π.

Approximating the hindsight distribution with a learned model $\hat{h}_k^\pi(a|s,u)$ and a reward model $\hat{r}(s,a)$, one can use an all-action policy gradient (Sutton et al., 2001) to update the actor's parameters θ,

$$\hat{\nabla} V_\pi(S_0) = \sum_{t\geq 0} \sum_{a\in\mathcal{A}} \nabla_\theta \pi_\theta(a|S_t) \left[\hat{r}(S_t,a) + \sum_{k\geq 1} \left(\frac{\hat{h}_k^\pi(a|S_t,U_{t+k})}{\pi_\theta(a|S_t)} \right) \gamma^k R_{t+k} \right]. \tag{2}$$

The outcome U_k can take different forms. In the original state-HCA formulation, $U_k = S_k$ (Harutyunyan et al., 2019). Meulemans et al. (2023) propose a general form of $U_k = f(S_k, A_k, R_k)$. As a specific instance, they introduce COCOA-reward, where $U_k = R_k$. To avoid learning a reward model, they also propose the following update:

$$\hat{\nabla} V_\pi(S_0) = \sum_{t\geq 0} \left[\nabla_\theta \log \pi_\theta(A_t|S_t) R_t + \sum_{a\in\mathcal{A}} \nabla_\theta \pi_\theta(a|S_t) \sum_{k\geq 1} \left(\frac{\hat{h}_k^\pi(a|S_t,U_{t+k})}{\pi_\theta(a|S_t)} \right) \gamma^k R_{t+k} \right]. \tag{3}$$

This update reduces to REINFORCE in the special case where $\hat{h}_k^\pi(a|S_t,U_{t+k}) = \mathbb{I}(A_t)$, where $I(\cdot)$ the indicator function placing all probability mass on the action that was actually taken, A_t.

Learning hindsight models. The success of HCA approaches depends on learning an accurate hindsight model ($\hat{h}_k^\pi$), trained via supervised learning to maximize the likelihood of actions given states and future outcomes. Inaccurate hindsight models may be the major factor limiting the broader applicability of these methods (Alipov et al., 2021). A key challenge is that the true hindsight distribution is policy-dependent. As the policy evolves, the action targets for each state-future pair shift, creating a non-stationary learning problem. Consequently, hindsight models are typically trained only on the latest interaction data and cannot reliably leverage replay buffers.

To mitigate this, previous work introduces a policy prior (Alipov et al., 2021; Meulemans et al., 2023). Instead of predicting the full hindsight distribution, the hindsight model learns only the correction relative to the current policy's output. Specifically, the hindsight model can be parameterized as $\hat{h}_k^\pi(a|s,u) = \text{softmax}\Big(g(s,u,k)\cdot f(s,u,k) + (1-g(s,u,k))\cdot \log \pi(a|s)\Big)$, where $f(s,u,k)$ produces (non-gated) predictions, and $g(s,u,k) \in [0,1]$ is a learned gate. When $g = 0$, the model defaults to the current policy, without using future information.

3 Illustrating the problem

Hindsight-based approaches are often motivated by settings where an early action is crucial to achieving a positive or negative outcome, with noisy intermediate steps in between. For instance, one can easily connect staying dry in an afternoon shower to the morning decision to carry an umbrella, despite the many unrelated events in between. The *umbrella problem*, commonly used to evaluate credit assignment capabilities (Osband et al., 2019), exemplifies this structure.

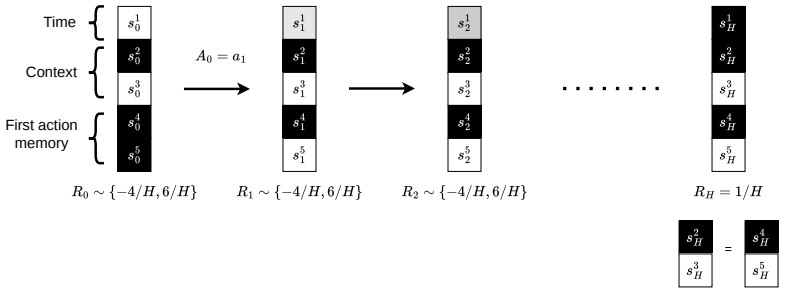


Figure 2: Illustration of the umbrella environment considered in Section 3. Black indicates the state component is zero, and white indicates that it is one. The agent receives noisy rewards at all steps except the final one. In the final step, the agent receives a positive reward if the consequence of the first action matches the context ($s_H^2 = s_H^4$).

In this section, we consider an episodic MDP based on the umbrella problem. We show that COCOA struggles in this environment even when using predictions from the ground-truth hindsight model. The failure stems from the REINFORCE-style term in the COCOA update (Equation 3), which handles immediate rewards without hindsight reweighting.

Environment details. The environment has a horizon H and two actions a_0, a_1. The state $s_t = (s_t^1, s_t^2, s_t^3, s_t^4, s_t^5)$ has five components. The first, $s_t^1 = 1 - t/H$, indicates time remaining. Components s_t^2, s_t^3 are complementary Boolean context indicators, set uniformly at random on reset and fixed throughout the episode. Components s_t^4, s_t^5 encode the first action: both are initially zero, then set to $(1, 0)$ if a_0 was taken and $(0, 1)$ otherwise. Actions have no influence after the first time step. At every non-terminal step, the agent receives a reward of $-4/H$ or $6/H$ with equal probability. On the final step, it receives $1/H$ if the context matches the action ($s_H^2 = s_H^4$) and $-1/H$ otherwise. We set the horizon $H = 50$ and the discount factor $\gamma = 1$ for our experiment. The environment is depicted in Figure 2.

Evaluation. We consider $U_k = R_k$, as in COCOA-reward (Meulemans et al., 2023). We train an actor network with data collected episodically using the *ground-truth* hindsight function, provided in Appendix D.

We evaluate agents by the probability they assign to the optimal action in the two possible starting states. Figure 3 shows that when both methods use the ground-truth hindsight model, COCOA learns significantly slower than R-HCA.

The failure of COCOA can be attributed to the first term in Equation 3, $\sum_{t \geq 0} \nabla_\theta \log \pi_\theta(A_t|S_t) R_t$, which handles immediate rewards through REINFORCE. In our umbrella environment with noisy rewards, this term dominates the overall sum [1]. Our approach, R-HCA, avoids this failure mode by incorporating immediate rewards into the hindsight reweighting.

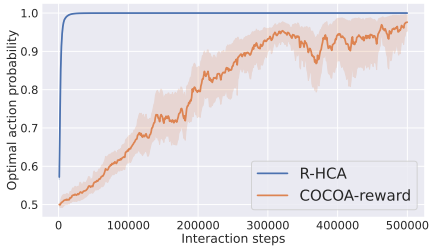


Figure 3: The probability that the policy assigns to the optimal action on the first time step in the Umbrella environment. Both agents use the ground-truth hindsight model. The solid line is the average over 10 seeds, and the shading denotes 95% bootstrap confidence intervals. COCOA learns significantly slower than R-HCA (Section 4).

4 Reward Hindsight Credit Assignment

In this section, we introduce Reward Hindsight Credit Assignment (R-HCA), a hindsight estimator that extends hindsight reweighting to incorporate immediate rewards. As illustrated in Section 3, COCOA handles immediate rewards through a REINFORCE-style term that can introduce high variance even

[1] This issue is less prominent in the similar key-to-door environment considered by Meulemans et al. (2023), where there is no noisy reward at the first step where the agent must pick up a key.

with a perfect hindsight model. We address this by modifying the COCOA update (Equation 3) to also re-weight immediate rewards through hindsight, yielding the R-HCA update:

$$\hat{\nabla} V_\pi(S_0) = \sum_{t \geq 0} \sum_{a \in \mathcal{A}} \nabla_\theta \pi(a|S_t) \sum_{k \geq 0} \left(\frac{\hat{h}_k^\pi(a|S_t, R_{t+k})}{\pi_\theta(a|S_t)} \right) \gamma^k R_{t+k}. \tag{4}$$

The difference from COCOA is that the sum over k now starts at $k = 0$. Now we prove that this is a valid policy gradient update. The proof is inspired by the work of Alipov et al. (2021), who show that a similar idea can be applied to *state*-HCA when rewards are a function of solely the next state. Our result does not require this assumption, as the hindsight model is conditioned on rewards directly.

Proposition 1. *Consider an action a and state s_t for which $\pi(a|s_t) > 0$. Then*

$$Q_\pi(s_t, a) = \mathbb{E}_{\tau \sim \mathcal{T}(s_t, \pi)} \left[\sum_{k \geq 0} \gamma^k \frac{h_k^\pi(a|s_t, R_{t+k})}{\pi(a|s_t)} R_{t+k} \right].$$

Proof. Expanding $Q_\pi(s_t, a) = \sum_{r \in \mathcal{R}} \sum_{k \geq 0} \gamma^k P(R_{t+k} = r|s_t, a)\, r$ and applying Bayes' rule to replace $P(R_{t+k} = r|s_t, a)$ with $\frac{P(a|s_t, R_{t+k}=r)\, P(R_{t+k}=r|s_t)}{P(a|s_t)}$, we identify the numerator's first factor as the hindsight distribution $h_k^\pi(a|s_t, r)$ and the denominator as $\pi(a|s_t)$. Converting the sum over r back to an expectation yields the result. □

The score-function policy gradient can be written as $\nabla_\theta V_\pi(S_0) = \mathbb{E}[\sum_t \nabla_\theta \log \pi_\theta(A_t|S_t) Q_\pi(S_t, A_t)]$. Substituting the hindsight expression for $Q_\pi(S_t, A_t)$ from Proposition 1 and converting to an all-action form (summing over actions a rather than sampling A_t) yields the R-HCA update in Equation 4. The full algorithm is presented in Algorithm 1.

COCOA and R-HCA differ only at $k = 0$. COCOA uses the REINFORCE estimator $\nabla_\theta \log \pi_\theta(A_t|S_t) R_t$, while R-HCA sums over all actions using hindsight weights, reducing variance from immediate rewards. To isolate the difference, we analyze them in the contextual bandit setting, where both estimators reduce to their $k = 0$ components.

Proposition 2. *Consider a stochastic contextual bandit defined by the following process: (1) A state (context) is sampled from the environment: $S \sim \rho(s)$ (2) Agent responds with an action sampled from the policy over a finite action set: $A \sim \pi_\theta(a|S)$ (3) A reward is sampled from the environment: $R \sim P_\mathcal{R}(r|S, A)$. Let $\hat{g}_{\text{COCOA}}$ be the gradient estimator used by COCOA (equivalent to REINFORCE in this setting) and $\hat{g}_{\text{R-HCA}}$ be the estimator used by R-HCA. Assuming a perfect hindsight model $\hat{h}(a|S, R) = P(A = a|S, R)$, the following variance decomposition holds:*

$$\mathit{Var}(\hat{g}_{\text{COCOA}}) = \mathit{Var}(\hat{g}_{\text{R-HCA}}) + \mathbb{E}[\mathit{Var}(\hat{g}_{\text{COCOA}} \mid S, R)].$$

In particular, $\mathit{Var}(\hat{g}_{\text{R-HCA}}) \leq \mathit{Var}(\hat{g}_{\text{COCOA}})$, due to non-negativity of variance.

Proof. The COCOA estimator depends on the full tuple of random variables (S, A, R). Using the definition from Equation 3 adapted to the single-step case ($k = 0$, no future terms)

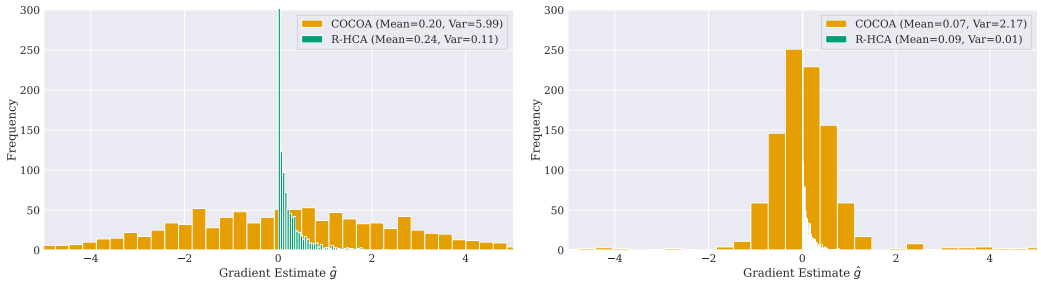
$$\hat{g}_{\text{COCOA}}(S, A, R) = \nabla_\theta \log \pi_\theta(A|S) R.$$

The R-HCA estimator, defined in Equation 4, integrates over the action space and therefore depends only on the random variables (S, R),

$$\hat{g}_{\text{R-HCA}}(S, R) = \sum_{a' \in \mathcal{A}} \nabla_\theta \pi_\theta(a'|S) \frac{\hat{h}(a'|S, R)}{\pi_\theta(a'|S)} R = R \sum_{a' \in \mathcal{A}} \hat{h}(a'|S, R) \nabla_\theta \log \pi_\theta(a'|S).$$

We condition on the outcome variables S and R and apply the Law of Total Variance:

$$\text{Var}(\hat{g}_{\text{COCOA}}) = \text{Var}(\mathbb{E}[\hat{g}_{\text{COCOA}} \mid S, R]) + \mathbb{E}[\text{Var}(\hat{g}_{\text{COCOA}} \mid S, R)].$$



(a) $\sigma(\theta) = 0.5$ (b) $\sigma(\theta) = 0.9$

Figure 4: Empirical distribution of gradient estimates for R-HCA and COCOA (equivalent to REINFORCE here) in a two-action Gaussian bandit with a perfect hindsight model, under two different policies (a) $\sigma(\theta) = 0.5$ and (b) $\sigma(\theta) = 0.9$. R-HCA has far lower variance, empirically confirming Proposition 2. Problem setup is described in Appendix C.

Conditioning on S and R fixes the state and reward, leaving A as the only source of randomness distributed according to the posterior $P(A|S, R)$. Since the hindsight model is perfect, $\hat{h}(a|S, R) = P(A = a|S, R)$, and thus:

$$\mathbb{E}[\hat{g}_{\text{COCOA}} \mid S, R] = \sum_{a \in \mathcal{A}} P(a|S, R) \nabla_\theta \log \pi_\theta(a|S) R = \hat{g}_{\text{R-HCA}}(S, R).$$

Substituting yields the decomposition. The inequality follows from the non-negativity of variance. □

We also empirically validate this variance reduction in a Gaussian bandit with a closed-form perfect hindsight model, where COCOA reduces to REINFORCE (Figure 4; Appendix C).

The variance reduction from Proposition 2 extends beyond contextual bandits. In Appendix A, we formally prove that R-HCA has lower variance than COCOA in sparse-reward MDPs, where non-zero rewards are only provided at the final state. In this setting, the immediate reward terms vanish at all non-terminal steps, and the terminal step reduces to a contextual bandit sub-problem where Proposition 2 applies directly.

For general MDPs, we provide a per-timestep variance decomposition in Appendix B. R-HCA always reduces the variance of the immediate reward term, but the overall comparison depends on the covariance between immediate reward differences and future returns. When this covariance is sufficiently negative, the immediate reward term in COCOA acts as a correlated control variate for future returns, and COCOA can have lower overall variance (see Appendix E for an explicit construction). However, as our experiments in Section 5 show, hindsight-based approaches are most beneficial when consequential actions are separated from informative outcomes by many timesteps of noisy intermediate rewards. In these settings, actions at non-consequential timesteps do not influence future returns, so the covariance term vanishes and R-HCA's guaranteed variance reduction (Proposition 2) dominates.

5 Experiments with learned hindsight models

In this section, we empirically evaluate R-HCA when the hindsight model is learned online alongside the policy [2]. Our experiments are designed to answer two questions: (1) Does R-HCA improve sample efficiency in long-term credit assignment problems when the hindsight model needs to be learned? (2) Under what conditions is reward-based hindsight credit assignment preferable to standard actor-critic methods?

[2]Code is available at https://github.com/Aditya-Ramesh-10/Reward-HCA

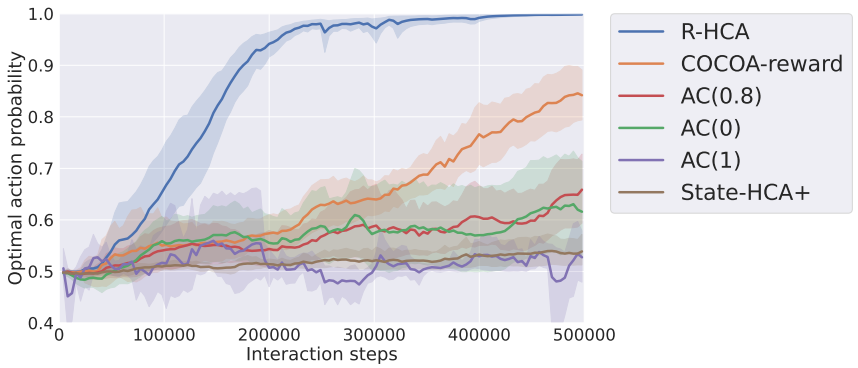


Figure 5: The probability that the policy assigns to the optimal action on the first time step in the umbrella environment. Hindsight-based approaches are trained with a learned hindsight model. R-HCA learns faster than all baselines.

First, we consider the Umbrella environment (introduced in Section 3), where only the very first action matters and all subsequent intermediate rewards are pure noise. Second, we test on a Planned Catch problem, where the difficulty of credit assignment is explicitly controlled by a delayed-consequence "lock-in" parameter and a noise level. Finally, we include two standard RL environments, CartPole and Lunar Lander, to test our method in domains where informative feedback closely follows actions.

In all performance plots, the solid line represents the average over 10 random seeds, and the shaded region denotes 95% bootstrap confidence intervals. Architecture specifications for the actor, critic, and hindsight model are provided in Appendix F.1.

5.1 Umbrella

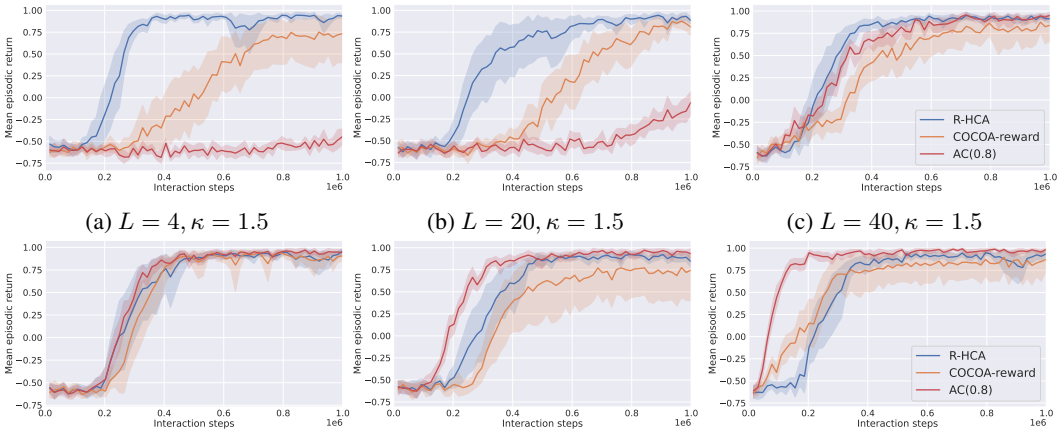
Recall from Section 3 that R-HCA efficiently solves this environment when given access to a ground-truth hindsight model (Figure 3). Here, we test a more realistic setting in which the hindsight model is learned concurrently with the actor.

Evaluation We compare R-HCA against COCOA-reward, A2C(λ) with $\lambda \in \{0, 0.8, 1\}$, and State-HCA+ (Meulemans et al., 2023) which uses the COCOA update (Equation 3) with $U_{t+k} = S_{t+k}$, conditioning on future states rather than future rewards. We perform a hyperparameter sweep over the actor-critic learning rate, entropy bonus coefficient, and the hindsight model learning rate (for R-HCA, COCOA, and State-HCA+). The full grid of considered and selected hyperparameters is detailed in Appendix F.2.

Results As shown in Figure 5, both reward-based hindsight approaches (R-HCA and COCOA-reward) successfully overcome noisy intermediate rewards and outperform the λ-return-based A2C baselines. Among the reward-based hindsight methods, R-HCA learns significantly faster than COCOA-reward. State-HCA+ fails to improve over the actor-critic baselines, consistent with the finding that conditioning on future states can cause the hindsight estimator to degenerate into REINFORCE (Meulemans et al., 2023). This confirms that the sample efficiency gains from R-HCA's variance reduction are not negated when using a learned hindsight model.

5.2 Planned Catch: When does R-HCA help?

The Umbrella environment tests a fixed, extreme credit assignment problem; here, we consider an environment where reward noise and the delay between actions and outcomes can be varied independently to compare all methods across a spectrum of credit assignment difficulty.



(a) $L = 4, \kappa = 1.5$ (b) $L = 20, \kappa = 1.5$ (c) $L = 40, \kappa = 1.5$

(d) $L = 4, \kappa = 0.5$ (e) $L = 20, \kappa = 0.5$ (f) $L = 40, \kappa = 0.5$

Figure 6: Mean episodic return in the evaluation version of the Planned Catch environment (no noisy rewards). We present results for lock-in $L \in \{4, 20, 40\}$ and for two levels of noise $\kappa \in \{1.5, 0.5\}$. R-HCA outperforms baselines in the hard credit assignment setting ($L = 4, \kappa = 1.5$) but lags behind the actor-critic in the easier settings ($L = 20, 40, \kappa = 0.5$).

Environment details The environment is a 40×5 grid ($H = 40$, $W = 5$). A fruit is initialized at a random column in the top row and descends one row per time step, yielding a fixed horizon of $T = H - 1$. The agent controls a paddle on the bottom row (initialized in the middle column) with actions $A = \{Left, Right, Stay\}$, bounded by the grid walls. A "catch" occurs when the agent and fruit share the same column at the final time step.

To study delayed consequences, we implement a lock-in mechanism. The agent's actions are effective only during the first L steps, after which the paddle is frozen. A small L forces the agent to commit early, increasing the delay between impactful actions and the outcome.

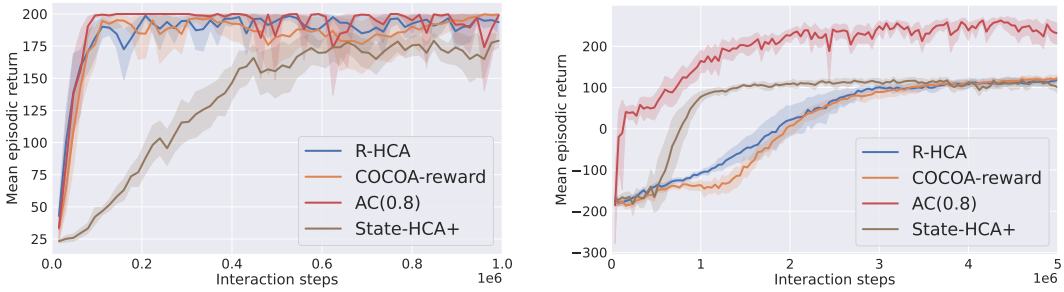
The reward is sparse and noisy. The agent receives $+1$ for catching the fruit and -1 for missing at the final step. At all intermediate steps, it receives a task-irrelevant reward sampled uniformly from $\{-\kappa, \kappa\}$.

Evaluation We compare R-HCA, COCOA, and A2C($\lambda = 0.8$) across two levels of noise, $\kappa = 1.5$ and $\kappa = 0.5$, and several values for the lock-in $L \in \{4, 8, 20, 32, 36, 40\}$. We report the average episodic return in a noise-free evaluation environment ($\kappa = 0$). Hyperparameter selection is described in Appendix F.3.

Results We present key results in Figure 6 and complete results for all considered choices of the lock-in parameter L in Appendix I (Figures 9 and 10).

In the high-noise setting (Figure 6, top row; $\kappa = 1.5$), R-HCA achieves strong performance across the full range of lock-in values, remaining roughly stable as L increases. A2C($\lambda = 0.8$) fails almost entirely under early lock-in (Figure 6a; $L = 4$), and only becomes competitive with R-HCA at late lock-in (Figure 6c; $L = 40$). With $L = 40$, informative feedback is only one step away from the last effective action, and TD-based credit propagation via λ-returns is well-suited to the problem. COCOA-reward also benefits from hindsight but consistently lags behind R-HCA across most values of L, suggesting that the high-variance immediate-reward issue identified in Section 3 impacts performance even in intermediate settings.

The low-noise setting (Figure 6, bottom row; $\kappa = 0.5$) shows that the advantage of hindsight-based methods is specifically tied to the combination of delayed consequences *and* noisy intermediate rewards. With sufficiently low noise, A2C($\lambda = 0.8$) is the strongest performer across nearly all values of L.



(a) CartPole

(b) Lunar Lander

Figure 7: Average episodic returns in CartPole and Lunar Lander.

Overall, we observe that R-HCA's performance is relatively insensitive to the lock-in parameter L and noise level κ. This robustness is an advantage in hard credit assignment settings, but it also means R-HCA cannot exploit the easier structure present when noise is low or feedback is temporally close. These results suggest that reward-based hindsight credit assignment is most valuable when consequential actions are separated from informative feedback by noisy intermediate rewards. When either the delay or the noise is reduced, standard actor-critic methods provide strong performance.

5.3 CartPole and Lunar Lander

The previous experiments demonstrate the advantages of R-HCA in environments where consequential actions are temporally separated from outcomes by noisy intermediate rewards. To further understand when standard actor-critic methods remain preferable, we evaluate on two standard RL environments, CartPole and Lunar Lander, where informative feedback closely follows actions. In CartPole, the agent receives a reward of +1 at every step the pole remains balanced. In Lunar Lander, shaped rewards for velocity, tilt, and leg contact provide dense feedback.

To maintain compatibility with our episodic setting (without value functions for hindsight-based methods), we modify the interactions with both environments to produce a fixed-length episode of 200 steps. If the task terminates early, the agent remains in the terminal state and receives zero reward for the remaining steps.

In CartPole (Figure 7a), A2C($\lambda = 0.8$) and the reward-based hindsight methods (R-HCA and COCOA-reward) all reach near-optimal performance, with A2C learning slightly faster. State-HCA+ converges significantly slower, suggesting difficulty in learning an accurate state-conditioned hindsight model in this environment.

In Lunar Lander (Figure 7b), A2C($\lambda = 0.8$) learns faster and achieves higher returns, while all hindsight-based methods lag far behind. Among the hindsight methods, R-HCA and COCOA-reward perform similarly. State-HCA+ learns faster than the reward-based hindsight methods but plateaus at a similar level, suggesting that state-conditioned hindsight is easier to learn in this environment but still insufficient to match TD-based credit propagation.

The reward in Lunar Lander depends on multiple factors such as velocity, tilt, and leg contact. We hypothesize that this makes it harder for the hindsight model to determine which action was responsible for a given reward. When hindsight models cannot reliably explain observed rewards, the reweighting becomes misleading, and standard actor-critic methods that propagate credit through a learned value function prove more effective.

6 Related Work

Our method extends the Hindsight Credit Assignment framework (HCA; Harutyunyan et al., 2019), building specifically on COCOA (Meulemans et al., 2023). Meulemans et al. (2023) investigate

which future features hindsight models should be conditioned on, proposing the use of future rewards or reward-encoding features. Our R-HCA algorithm differs from COCOA in how it handles credit associated with immediate rewards.

Several other works propose modifications to the original HCA formulation. One suggestion is to redistribute TD-errors instead of rewards to improve State-HCA (Young, 2019; Alipov et al., 2021), while another work designs a more stable method for estimating the hindsight ratio (Velu et al., 2023). These complementary ideas could be applied with our R-HCA algorithm.

HCA also shares similarities with upside-down reinforcement learning (UDRL; Schmidhuber, 2019; Srivastava et al., 2019) and related approaches (Kumar et al., 2019; Chen et al., 2021), which typically learn a hindsight model conditioned on future returns. However, UDRL uses this model to directly select actions, whereas HCA maintains a separate actor and uses the hindsight model's predictions to improve advantage estimation.

More broadly, HCA can be viewed as a model-based approach to credit assignment. Other model-based methods for credit assignment include differentiating through a learned model (Werbos, 1987; Schmidhuber, 1990; Heess et al., 2015) and using models to compress trajectories (Ramesh et al., 2024). In contrast to these approaches, which rely on a forward-dynamics model, HCA leverages a multi-step inverse dynamics model to associate rewards with state-action pairs.

Related credit assignment methods involve learning future-conditioned baselines or critics (Mesnard et al., 2021; Nota et al., 2021; Mesnard et al., 2023). These critic-based methods are complementary to R-HCA. Proposition 1 expresses the action value as a hindsight-reweighted return, which can be used as a target to train a parametric critic. We leave this combination to future work.

RUDDER (Arjona-Medina et al., 2019) and related methods (Patil et al., 2022) learn a forward return predictor to retrospectively reshape rewards, rather than reweighting the policy gradient through a hindsight model like R-HCA. Other approaches to credit assignment include identifying critical steps in trajectories (Sun et al., 2023) and predicting synthetic reward contributions in advance (Raposo et al., 2021). Finally, hindsight re-labeling has proven effective in goal-conditioned RL (Andrychowicz et al., 2017; Rauber et al., 2019). These methods retroactively substitute intended goals with achieved ones to more efficiently use collected data. Our approach instead uses future information to infer which past actions were responsible for observed rewards, targeting credit assignment in general reward-based RL.

7 Conclusion

In this paper, we introduce R-HCA, a credit assignment algorithm that learns a hindsight model to associate future rewards with previous states and actions. Our method addresses a critical limitation of the previously proposed COCOA, which degenerates into a high-variance estimator in environments with noisy immediate rewards. We propose an estimator that incorporates immediate rewards through hindsight re-weighting. We prove that the variance of the R-HCA estimator is no greater than that of COCOA in contextual bandits and sparse-reward MDPs. For general MDPs, our variance decomposition shows that COCOA can have lower variance than R-HCA in certain settings. However, our experiments show that R-HCA is the stronger choice in the settings where hindsight-based approaches are most needed, namely environments where consequential actions are separated from informative feedback by many timesteps of noisy intermediate rewards.

While hindsight-based approaches show promise, their success depends on learning an accurate hindsight model, which must be retrained as the policy evolves. Our results on Lunar Lander show that reward-based hindsight credit assignment can underperform standard actor-critic methods when the reward structure is difficult for the hindsight model to explain. Developing hindsight models that are more robust to policy non-stationarity remains a key challenge for making these methods more broadly applicable. Hybrid approaches that selectively apply hindsight reweighting only at timesteps where it is likely to help, while relying on standard TD-based credit assignment elsewhere, could combine the strengths of both paradigms.

A R-HCA has lower variance than COCOA in Sparse Reward MDPs

Proposition 3. *Consider an episodic Markov Decision Process (MDP) of horizon H with sparse rewards, where the agent receives a reward $R_t = 0$ for all $t < H$, and a terminal reward $R_H \in \mathcal{R}$ at the final step. Let $\hat{g}_{COCOA}$ and $\hat{g}_{R-HCA}$ be the full-episode policy gradient estimators defined by COCOA-reward (Equation 3) and R-HCA (Equation 4), respectively. Assuming a perfect hindsight model where $\hat{h}^{\pi}_{k}(a|s,r) = P(A_t = a|S_t = s, R_{t+k} = r)$, the variance of the R-HCA estimator is bounded by the variance of the COCOA-reward estimator:*

$$Var(\hat{g}_{R-HCA}) \leq Var(\hat{g}_{COCOA})$$

Proof. We decompose the full-episode estimators as $\hat{g} = \sum_{t=0}^{H} \hat{g}_t$, where each per-timestep estimator splits into an immediate reward term ($k = 0$) and a future reward term ($k \geq 1$). The future reward term $G_{\text{fut},t}$ is identical for both algorithms at every timestep.

Case 1: Non-terminal steps ($t < H$). Since $R_t = 0$ by assumption, the immediate reward terms for both estimators vanish, and thus $\hat{g}_{COCOA,t} = \hat{g}_{R-HCA,t} = G_{\text{fut},t}$.

Case 2: Terminal step ($t = H$). At the final step, there are no future rewards, so $G_{\text{fut},H} = 0$ and both estimators reduce entirely to their immediate reward terms based on R_H. Since $A_H \sim \pi(\cdot|S_H)$ and R_H is a function of (S_H, A_H) only, we have $P(A_H \mid S_H, R_H, \text{history}) = P(A_H \mid S_H, R_H)$ by the Markov property. The terminal step therefore defines a contextual bandit sub-problem, and by the same argument as Proposition 2, $\text{Var}(\hat{g}_{R-HCA,H}) \leq \text{Var}(\hat{g}_{COCOA,H})$.

Conclusion. Let $\Sigma_{<H} = \sum_{t=0}^{H-1} \hat{g}_t$ denote the sum of gradients for non-terminal steps, which are identical for both estimators. Expanding the full-episode variance:

$$\text{Var}(\hat{g}_{COCOA}) = \text{Var}(\Sigma_{<H}) + \text{Var}(\hat{g}_{COCOA,H}) + 2\,\text{Cov}(\Sigma_{<H}, \hat{g}_{COCOA,H})$$
$$\text{Var}(\hat{g}_{R-HCA}) = \text{Var}(\Sigma_{<H}) + \text{Var}(\hat{g}_{R-HCA,H}) + 2\,\text{Cov}(\Sigma_{<H}, \hat{g}_{R-HCA,H})$$

Let $D_H = \hat{g}_{COCOA,H} - \hat{g}_{R-HCA,H}$. Since A_H is independent of the trajectory history given (S_H, R_H), we have $\text{Cov}(\Sigma_{<H}, D_H \mid S_H, R_H) = 0$; combined with $\mathbb{E}[D_H|S_H, R_H] = 0$ (by the same argument as Proposition 2), the Law of Total Covariance gives $\text{Cov}(\Sigma_{<H}, D_H) = 0$, so the covariance terms are identical across both estimators. Together with $\text{Var}(\hat{g}_{R-HCA,H}) \leq \text{Var}(\hat{g}_{COCOA,H})$, we conclude $\text{Var}(\hat{g}_{R-HCA}) \leq \text{Var}(\hat{g}_{COCOA})$. □

B Variance decomposition for R-HCA vs COCOA

In Proposition 2 (Section 4), we showed that R-HCA has lower variance than COCOA for the immediate reward term, corresponding to the contextual bandit setting. In Proposition 3 (Appendix A), we extended this to the full-episode estimator in sparse-reward MDPs. Here, we characterize the variance comparison in general MDPs, where the relationship between the two estimators depends on the interaction between immediate and future reward terms.

Proposition 4. *Let $\hat{g}_{\text{COCOA},t}$ and $\hat{g}_{\text{R-HCA},t}$ be the per-timestep policy gradient estimators defined by COCOA-reward (Equation 3) and R-HCA (Equation 4), respectively. Assuming a perfect hindsight model where $\hat{h}^{\pi}_{k}(a|s,r) = P(A_t = a \mid S_t = s, R_{t+k} = r)$, the variance difference admits the decomposition:*

$$Var(\hat{g}_{\text{COCOA},t}) - Var(\hat{g}_{\text{R-HCA},t}) = \underbrace{[Var(I_{\text{COCOA},t}) - Var(I_{\text{R-HCA},t})]}_{\geq 0 \text{ (Proposition 2)}} + 2\,Cov(D_t,\, G_{fut,t}) \tag{5}$$

where $I_{\text{COCOA},t} = \nabla_\theta \log \pi_\theta(A_t|S_t)\, R_t$ is the COCOA immediate reward term, $I_{\text{R-HCA},t} = R_t \sum_{a \in \mathcal{A}} \hat{h}^{\pi}_{0}(a|S_t, R_t)\, \nabla_\theta \log \pi_\theta(a|S_t)$ is the R-HCA immediate reward term,

$$G_{fut,t} = \sum_{a \in \mathcal{A}} \nabla_\theta \log \pi_\theta(a|S_t) \sum_{k \geq 1} \hat{h}^{\pi}_{k}(a|S_t, R_{t+k})\, \gamma^k R_{t+k}$$

is the future reward term shared by both estimators, and $D_t = I_{COCOA,t} - I_{R\text{-}HCA,t}$.

Consequently, $Var(\hat{g}_{R\text{-}HCA,t}) \leq Var(\hat{g}_{COCOA,t})$ *if and only if*

$$Var(I_{COCOA,t}) - Var(I_{R\text{-}HCA,t}) \;\geq\; -2\,Cov(D_t,\, G_{fut,t}). \tag{6}$$

Proof. Both estimators decompose into an immediate reward term ($k = 0$) and a future reward term ($k \geq 1$):

$$\hat{g}_{\text{COCOA},t} = I_{\text{COCOA},t} + G_{\text{fut},t}, \qquad \hat{g}_{\text{R-HCA},t} = I_{\text{R-HCA},t} + G_{\text{fut},t},$$

where $G_{\text{fut},t}$ is identical for both estimators (as established in Section 4). Expanding the variance of each:

$$\begin{aligned}
\text{Var}(\hat{g}_{\text{COCOA},t}) &= \text{Var}(I_{\text{COCOA},t}) + \text{Var}(G_{\text{fut},t}) + 2\,\text{Cov}(I_{\text{COCOA},t},\, G_{\text{fut},t}),\\
\text{Var}(\hat{g}_{\text{R-HCA},t}) &= \text{Var}(I_{\text{R-HCA},t}) + \text{Var}(G_{\text{fut},t}) + 2\,\text{Cov}(I_{\text{R-HCA},t},\, G_{\text{fut},t}).
\end{aligned}$$

Subtracting and applying linearity of covariance:

$$\begin{aligned}
\text{Var}(\hat{g}_{\text{COCOA},t}) - \text{Var}(\hat{g}_{\text{R-HCA},t}) &= [\text{Var}(I_{\text{COCOA},t}) - \text{Var}(I_{\text{R-HCA},t})]\\
&\quad + 2\,[\text{Cov}(I_{\text{COCOA},t},\, G_{\text{fut},t}) - \text{Cov}(I_{\text{R-HCA},t},\, G_{\text{fut},t})]\\
&= [\text{Var}(I_{\text{COCOA},t}) - \text{Var}(I_{\text{R-HCA},t})] + 2\,\text{Cov}(D_t,\, G_{\text{fut},t}).
\end{aligned}$$

The first term is non-negative by Proposition 2: conditioned on S_t, the immediate reward terms define a contextual bandit sub-problem in which R-HCA has lower or equal variance. □

The decomposition in Equation 5 has two terms. The first term is always non-negative (Proposition 2): R-HCA reduces the variance of the immediate reward contribution by filtering out noise from the immediate reward term that COCOA retains.

The second term, $\text{Cov}(D_t, G_{\text{fut},t})$, can be negative. COCOA's immediate term $I_{\text{COCOA},t}$ depends on the sampled action A_t, as does $G_{\text{fut},t}$ through the effect of A_t on future outcomes.

Whether R-HCA has lower variance than COCOA depends on which effect dominates. In the Umbrella environment (Section 3), for instance, only the action at $t = 0$ affects future rewards. At all subsequent timesteps, $G_{\text{fut},t} = 0$, so the covariance term vanishes and R-HCA eliminates COCOA's noisy immediate term at zero cost. At $t = 0$, the future reward term is nonzero but small, so the covariance term is negligible compared to the cumulative noise reduction across the other $H - 1$ timesteps. More generally, whenever consequential decisions and uninformative immediate rewards are separated across timesteps, R-HCA benefits from variance reduction at the non-consequential steps without paying a significant cost at the consequential ones.

In the special cases of contextual bandits ($G_{\text{fut},t} = 0$) and sparse-reward MDPs ($R_t = 0$ for non-terminal t), the covariance term vanishes entirely and R-HCA is guaranteed to have lower or equal variance (Propositions 2 and 3, respectively).

C Gaussian bandit problem

We empirically compare the gradient variance of R-HCA and COCOA (equivalent to REINFORCE here) in a two-action Gaussian bandit. The agent selects $A \sim \pi_\theta$ via a scalar logit θ, with $\pi_\theta(a_1) = \sigma(\theta) = 1/(1 + e^{-\theta})$ and $\pi_\theta(a_0) = 1 - \sigma(\theta)$. The reward is $R \sim \mathcal{N}(\mu_A, \sigma^2)$, with $\mu_{a_0} = -0.5$, $\mu_{a_1} = 0.5$, and $\sigma = 5.0$.

Since $\nabla_\theta \log \pi_\theta(a_1) = 1 - \sigma(\theta)$ and $\nabla_\theta \log \pi_\theta(a_0) = -\sigma(\theta)$, the two estimators are

$$\hat{g}_{\text{COCOA}} = \begin{cases} -\sigma(\theta)R & \text{if } A = a_0 \\ (1 - \sigma(\theta))R & \text{if } A = a_1 \end{cases} \qquad \hat{g}_{\text{R-HCA}} = -\hat{h}(a_0 \mid R)\sigma(\theta)R + \hat{h}(a_1 \mid R)(1 - \sigma(\theta))R.$$

We sample $N = 1000$ interactions and evaluate the gradients under the perfect hindsight model, $\hat{h}(a \mid r) = P(a \mid r)$. Figure 4 shows the resulting distributions for $\sigma(\theta) \in \{0.5, 0.9\}$; consistent with Proposition 2, R-HCA substantially reduces variance.

References

V.M. Aleksandrov, V.I. Sysoyev, and V.V. Shemenev. Stochastic optimization. *Engineering Cybernetics*, (5):11–16, 1968.

Vyacheslav Alipov, Riley Simmons-Edler, Nikita Putintsev, Pavel Kalinin, and Dmitry Vetrov. Towards practical credit assignment for deep reinforcement learning. *arXiv preprint arXiv:2106.04499*, 2021.

S. Amari. A theory of adaptive pattern classifiers. *IEEE Trans. EC*, 16(3):299–307, 1967.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

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.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR*, 2015.

Richard Bellman. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6):503–515, 1954.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Brett Daley, Marlos C. Machado, and Martha White. Demystifying the recency heuristic in temporal-difference learning. *Reinforcement Learning Journal (RLJ)*, 3:1019–1036, 2024.

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*, 32:12488–12497, 2019.

Nicolas Heess, Gregory Wayne, David Silver, Timothy Lillicrap, Tom Erez, and Yuval Tassa. Learning continuous control policies by stochastic value gradients. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Ehsan Imani and Martha White. Improving regression performance with distributional losses. In *International conference on machine learning*, pp. 2157–2166. PMLR, 2018.

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

Aviral Kumar, Xue Bin Peng, and Sergey Levine. Reward-conditioned policies. *arXiv preprint arXiv:1912.13465*, 2019.

Thomas Mesnard, Theophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Thomas S Stepleton, Nicolas Heess, Arthur Guez, et al. Counterfactual credit assignment in model-free reinforcement learning. In *International Conference on Machine Learning*, pp. 7654–7664. PMLR, 2021.

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

Alexander Meulemans, Simon Schug, Seijin Kobayashi, Nathaniel Daw, and Greg Wayne. Would i have gotten that reward? long-term credit assignment by counterfactual contribution analysis. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pp. 1928–1937. PmLR, 2016.

Chris Nota, Philip Thomas, and Bruno C Da Silva. Posterior value functions: Hindsight baselines for policy gradient methods. In *International Conference on Machine Learning*, pp. 8238–8247. PMLR, 2021.

Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepesvari, Satinder Singh, et al. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568*, 2019.

Vihang Patil, Markus Hofmarcher, Marius-Constantin Dinu, Matthias Dorfer, Patrick M Blies, Johannes Brandstetter, José Arjona-Medina, and Sepp Hochreiter. Align-rudder: Learning from few demonstrations by reward redistribution. In *International Conference on Machine Learning*, pp. 17531–17572. PMLR, 2022.

Eduardo Pignatelli, Johan Ferret, Matthieu Geist, Thomas Mesnard, Hado van Hasselt, and Laura Toni. A survey of temporal credit assignment in deep reinforcement learning. *Trans. Mach. Learn. Res.*, 2024.

Aditya A. Ramesh, Kenny John Young, Louis Kirsch, and Jürgen Schmidhuber. Sequence compression speeds up credit assignment in reinforcement learning. In *Forty-first International Conference on Machine Learning, ICML*, 2024.

David Raposo, Sam Ritter, Adam Santoro, Greg Wayne, Theophane Weber, Matt Botvinick, Hado van Hasselt, and Francis Song. Synthetic returns for long-term credit assignment. *arXiv preprint arXiv:2102.12425*, 2021.

Paulo E. Rauber, Avinash Ummadisingu, Filipe Mutz, and Jürgen Schmidhuber. Hindsight policy gradients. In *7th International Conference on Learning Representations, ICLR*, 2019.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Jürgen Schmidhuber. Making the world differentiable: On using fully recurrent self-supervised neural networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report FKI-126-90, Tech. Univ. Munich, 1990.

Jürgen Schmidhuber. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 4(1):131–139, 1992.

Jürgen Schmidhuber. Reinforcement learning upside down: Don't predict rewards–just map them to actions. *arXiv preprint arXiv:1912.02875*, 2019.

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.

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

Rupesh Kumar Srivastava, Pranav Shyam, Filipe Mutz, Wojciech Jaśkowski, and Jürgen Schmidhuber. Training agents using upside-down reinforcement learning. *arXiv preprint arXiv:1912.02877*, 2019.

Chen Sun, Wannan Yang, Thomas Jiralerspong, Dane Malenfant, Benjamin Alsbury-Nealy, Yoshua Bengio, and Blake Richards. Contrastive retrospection: honing in on critical steps for rapid learning and generalization in rl. *Advances in Neural Information Processing Systems*, 36:31117–31139, 2023.

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

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

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Richard S Sutton, Satinder Singh, and David McAllester. Comparing policy-gradient algorithms. *Technical Report*, 2001.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Akash Velu, Skanda Vaidyanath, and Dilip Arumugam. Hindsight-dice: Stable credit assignment for deep reinforcement learning. *arXiv preprint arXiv:2307.11897*, 2023.

Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

P. J. Werbos. Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research. *IEEE Transactions on Systems, Man, and Cybernetics*, 17, 1987.

R Williams. A class of gradient-estimation algorithms for reinforcement learning in neural networks. In *Proceedings of the International Conference on Neural Networks*, 1987.

Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

Kenny Young. Variance reduced advantage estimation with δ-hindsight credit assignment. *arXiv preprint arXiv:1911.08362*, 2019.

Supplementary Materials

The following content was not necessarily subject to peer review.

D Ground-truth hindsight model for the umbrella problem

In the umbrella environment (Section 3), the only action that influences the final outcome is the one taken at the very first timestep. With u as future rewards, the ground-truth hindsight model can be specified as:

$$h_k^\pi(s,u) = \begin{cases} \mathbb{I}(a_1) & \text{if} \quad u = 1/H, s^4 = 0, s^5 = 0, s^3 = 1, s^2 = 0, \\ \mathbb{I}(a_1) & \text{if} \quad u = -1/H, s^4 = 0, s^5 = 0, s^3 = 0, s^2 = 1, \\ \mathbb{I}(a_0) & \text{if} \quad u = 1/H, s^4 = 0, s^5 = 0, s^3 = 0, s^2 = 1, \\ \mathbb{I}(a_0) & \text{if} \quad u = -1/H, s^4 = 0, s^5 = 0, s^3 = 1, s^2 = 0, \\ \pi(s) & \text{otherwise.} \end{cases}$$

E R-HCA can have a larger variance than COCOA

Example 1. Consider an episodic MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, R, \gamma \rangle$ as illustrated in Figure 8, with a deterministic initial state S_0 and discount factor $\gamma = 1$. Let the action space at S_0 be $\mathcal{A} = \{a_0, a_1, a_2\}$.

Let the policy be parameterized by a single scalar $\theta \in \mathbb{R}$ via a softmax distribution over action logits $h_\theta(a)$:

$$\pi_\theta(a|S_0) = \frac{\exp(h_\theta(a))}{\sum_{a' \in \mathcal{A}} \exp(h_\theta(a'))}$$

where the logits are defined as $h_\theta(a_0) = -\theta$, $h_\theta(a_1) = 0$, and $h_\theta(a_2) = \theta$. We evaluate the gradient at $\theta = 0$. At this parameterization, the policy is uniform ($\pi_0(a|S_0) = 1/3$ for all $a \in \mathcal{A}$), and the expected logit gradient is exactly zero. This yields the following explicit score function values at S_0:

$$\nabla_\theta \log \pi_\theta(a_0|S_0)\Big|_{\theta=0} = -1, \quad \nabla_\theta \log \pi_\theta(a_1|S_0)\Big|_{\theta=0} = 0, \quad \nabla_\theta \log \pi_\theta(a_2|S_0)\Big|_{\theta=0} = 1.$$

Taking any action at S_0 yields a deterministic immediate step-cost of $R_0 = 1$ and transitions the agent to one of three intermediate states: S_1, S_2, or S_3. At step $t = 1$, the agent has only a single available action a, meaning the policy is deterministic ($\pi(a|S_i) = 1$) and introduces no further gradient variance ($\nabla_\theta \log \pi(a|S_i) = 0$).

The transitions from the intermediate states to the terminal states determine the final goal reward R_1:

- State S_1 transitions deterministically to yield $R_1 = -5$.
- State S_3 transitions deterministically to yield $R_1 = -15$.
- State S_2 transitions stochastically, yielding $R_1 = -5$ with probability 0.5 and $R_1 = -15$ with probability 0.5.

Let $\hat{g}_{\text{COCOA}}$ denote the COCOA estimator and $\hat{g}_{\text{R-HCA}}$ denote the R-HCA estimator where the immediate action A_0 is analytically summed out given the true posterior $P(A_0|R_1)$. Because R_0 is constant and the step $t = 1$ gradients are zero, the marginalized immediate reward term in $\hat{g}_{\text{R-HCA}}$ vanishes ($\sum_a \pi_\theta(a|S_0) \nabla_\theta \log \pi_\theta(a|S_0) R_0 = 0$). The estimators for the entire episode thus simplify

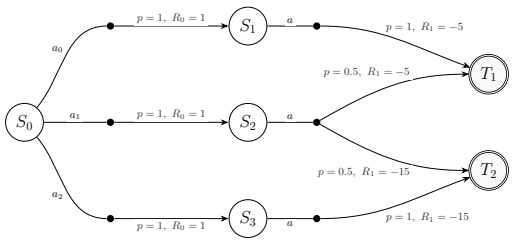


Figure 8: An MDP counterexample in which R-HCA could have a larger variance than COCOA. See Example 1 for details.

to:

$$\hat{g}_{\text{COCOA}} = \nabla_\theta \log \pi_\theta(A_0|S_0)R_0 + \hat{g}_{\text{R-HCA}}$$
$$\hat{g}_{\text{R-HCA}} = \sum_{a \in \mathcal{A}} \nabla_\theta \log \pi_\theta(a|S_0)P(A_0 = a|R_1)R_1.$$

Evaluating the posterior $P(A_0 = a|R_1)$ for the two equiprobable outcomes $R_1 \in \{-5, -15\}$ shows that $\hat{g}_{\text{R-HCA}}$ takes the following values:

1. $R_1 = -5$ with probability $1/2$: $\hat{g}_{\text{R-HCA}} = [-1 \times 2/3 + 0 \times 1/3 + 1 \times 0] \times (-5) = 10/3$;
2. $R_1 = -15$ with probability $1/2$: $\hat{g}_{\text{R-HCA}} = [-1 \times 0 + 0 \times 1/3 + 1 \times 2/3] \times (-15) = -10$.

The variance of $\hat{g}_{\text{R-HCA}}$ is:

$$\text{Var}(\hat{g}_{\text{R-HCA}}) = \frac{400}{9} \approx 44.44.$$

Conversely, evaluating $\hat{g}_{\text{COCOA}}$ over the four possible trajectory realizations (A_0, R_1) yields:

1. $(A_0, R_1) = (a_0, -5)$ with probability $1/3$: $\hat{g}_{\text{COCOA}} = -1 \times 1 + 10/3 = 7/3$;
2. $(A_0, R_1) = (a_1, -5)$ with probability $1/6$: $\hat{g}_{\text{COCOA}} = 0 \times 1 + 10/3 = 10/3$;
3. $(A_0, R_1) = (a_1, -15)$ with probability $1/6$: $\hat{g}_{\text{COCOA}} = 0 \times 1 - 10 = -10$;
4. $(A_0, R_1) = (a_2, -15)$ with probability $1/3$: $\hat{g}_{\text{COCOA}} = 1 \times 1 - 10 = -9$.

Calculating the variance of $\hat{g}_{\text{COCOA}}$ across these outcomes yields:

$$\text{Var}(\hat{g}_{\text{COCOA}}) = \frac{326}{9} \approx 36.22.$$

Since $44.44 > 36.22$, we conclude that $\text{Var}(\hat{g}_{\text{R-HCA}}) > \text{Var}(\hat{g}_{\text{COCOA}})$. This demonstrates that analytic marginalization of the immediate action can strictly increase estimator variance when the immediate step-cost acts as a negatively correlated control variate for future goal rewards.

F Implementation details

F.1 Neural network architectures

Actor-Critic The actor and critic are independent Multi-Layer Perceptrons (MLPs) that share no parameters. Both networks take flattened observations as input and process them through a body of 3 hidden layers, each with 512 units and ReLU non-linearities.

- **Actor:** The actor head outputs the logits of a categorical distribution over the discrete action space.
- **Critic:** The critic head outputs a scalar state-value estimate.

- **Initialization:** We apply orthogonal weight initialization across all layers. The hidden layers use a gain of $\sqrt{2}$ (standard for ReLU activations). To ensure an initially near-uniform policy, the actor's final layer uses a gain of 0.01, while the critic's final layer uses a gain of 1.0. All biases are initialized to a constant of 0.0.

Note that the hindsight-based approaches (R-HCA and COCOA) do not require a critic in the episodic settings considered here. For these methods, we disable learning in the critic by setting the value loss coefficient to zero.

Hindsight Model The hindsight model ($\hat{h}^{\pi}_{k}(\cdot|s_t, r_{t+k})$) takes as input a state, future reward, policy log-probabilities at the state, and the time elapsed between the state and reward. The model utilizes a scaled dot-product interaction mechanism to compute non-gated predictions, which are then combined with the policy log-probabilities via a learned gating mechanism. We set the base embedding size to $d = 512$ and the interaction dimension to $d_{int} = 128$.

State Encoding The state observation is flattened and linearly projected to a vector in $\mathbb{R}^d$. This feature vector is passed through a residual network block consisting of three linear layers with ReLU activations and Layer Normalization. Finally, the features are linearly projected to a dimension of $d_{int} \times |\mathcal{A}|$, where $|\mathcal{A}|$ is the number of possible actions.

Reward and Time-Difference Encoding The scalar future reward is modified into a "two-hot" categorical distribution (Imani & White, 2018) using 51 bins between r_{min} and r_{max}. This two-hot representation is linearly encoded to a d-dimensional vector. We additively combine this with a learned positional embedding corresponding to the time elapsed (k). Similar to the state encoding, this combined vector passes through a residual MLP block (with ReLU and Layer Normalization) and is linearly projected to $d_{int} \times |\mathcal{A}|$.

Dot-Product Interaction (Non-gated Predictions) To compute the non-gated hindsight predictions, we reshape the state and future-info projections into tensors of shape $(d_{int}, |\mathcal{A}|)$. We perform an element-wise multiplication between the two and sum across the d_{int} dimension. The result is scaled by $1/\sqrt{d_{int}}$ to stabilize gradients, yielding a prediction vector of size $|\mathcal{A}|$.

Gating Mechanism To handle the non-stationarity of the changing policy, the hindsight model predicts a residual correction to the policy's outputs via a learned gate. We use a coupled gate that outputs a single scalar $g \in [0, 1]$. We reuse the d-dimensional state and future-info features from earlier in the network (detached from the main computational graph), concatenate them, and pass them through a two-layer MLP with a ReLU activation, followed by a sigmoid, to produce g.

The final hindsight prediction is obtained by a convex combination of the non-gated predictions and the centered policy log-probabilities.

Initialization The initial linear encoders for the state and future reward use orthogonal initialization with a gain of 1.0. All subsequent linear layers, interaction projections, and gating projections are initialized using a normal distribution with a standard deviation of 0.02.

Optimization The actor and critic networks are trained with the RMSProp optimizer. The hindsight models are trained with the Adam optimizer.

General hyperparameters common to all approaches are provided in Table 1. Hyperparameters for each environment are provided in the subsequent sections.

F.2 Umbrella

We consider different choices of the learning rate (LR) and entropy bonus coefficient (EB). On-policy actor-critic approaches are commonly trained with an entropy bonus to prevent premature convergence to a deterministic policy (Williams & Peng, 1991; Mnih et al., 2016). We consider EB $\in \{0.0001, 0.001, 0.01\}$ for all methods. For the actor (and critic in the case of AC), we consider LR $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3, 5e{-}3\}$ for AC vari-

Table 1: Common hyperparameters to all experiments

Hyperparameter	Value
Discount factor	1.0
Rollout length (frames per process)	Episodic
Parallel processes	4
Value loss coefficient	0.5 (0 for hindsight methods)
Hindsight-ratio max clipping	10 (for hindsight methods)

ants and LR $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3\}$ for the hindsight-based methods. For R-HCA, COCOA, and State-HCA+, we also consider the hindsight model learning rate LRH $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4\}$ and number of updates to the hindsight model per batch $K \in \{1, 2\}$. For the two-hot encoding of rewards we use $r_{min} = -0.1$ and $r_{max} = 0.1$. The selected values are reported in Tables 2 and 3.

Table 2: Selected hyperparameters for the actor (and critic in the case of A2C) in the umbrella problem.

Algorithm	Selected LR	Selected EB
A2C(0)	$1e{-}4$	$1e{-}3$
A2C(0.8)	$1e{-}4$	$1e{-}4$
A2C(1)	$5e{-}4$	$1e{-}3$
COCOA-reward	$1e{-}4$	$1e{-}4$
State-HCA+	$1e{-}4$	$1e{-}3$
R-HCA	$5e{-}5$	$1e{-}4$

Table 3: Selected hyperparameters for the hindsight model in the umbrella problem.

Algorithm	Selected LRH	Selected K
COCOA-reward	$5e-5$	2
State-HCA+	$1e-4$	2
R-HCA	$5e-5$	2

F.3 Planned Catch

We perform a hyperparameter sweep over the actor-critic learning rate (LR), entropy bonus coefficient (EB), and (for R-HCA and COCOA) the hindsight model learning rate (LRH). We consider LR $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3\}$, EB $\in \{0.1, 0.01, 0.001\}$, and LRH $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4\}$. We update the hindsight model once per batch ($K = 1$). For the two-hot encoding of rewards we use $r_{min} = -1.5$ and $r_{max} = 1.5$. Hyperparameters are selected using settings $L = 4$ and $L = 40$ with $\kappa = 1.5$, and the same selected values are used for all values of lock-in L and noise level κ. The selected values are reported in Tables 4 and 5.

F.4 CartPole

We follow the same hyperparameter sweep protocol as in the other environments. We consider LR $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3\}$, EB $\in \{0.1, 0.01, 0.001\}$, LRH $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4\}$, and number of updates to the hindsight model per batch $K \in \{1, 2\}$. For the two-hot encoding of rewards we use $r_{min} = 0$ and $r_{max} = 1$. The selected values are reported in Tables 6 and 7.

Table 4: Selected hyperparameters for the actor (and critic in the case of A2C) in Planned Catch.

Algorithm	Selected LR	Selected EB
A2C(0.8)	$5e{-}4$	0.1
COCOA-reward	$5e{-}4$	0.1
R-HCA	$1e{-}4$	0.1

Table 5: Selected hyperparameters for the hindsight model in Planned Catch.

Algorithm	Selected LRH
COCOA-reward	$1e{-}4$
R-HCA	$1e{-}4$

F.5 Lunar Lander

We follow the same hyperparameter sweep protocol as in the other environments. We consider LR $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3\}$, EB $\in \{0.1, 0.01, 0.001\}$, LRH $\in \{1e{-}5, 5e{-}5, 1e{-}4, 5e{-}4\}$, and number of updates to the hindsight model per batch $K \in \{1, 2\}$. For the two-hot encoding of rewards we use $r_{min} = -100$ and $r_{max} = 100$. The selected values are reported in Tables 8 and 9.

Table 6: Selected hyperparameters for the actor and critic in CartPole.

Algorithm	Selected LR	Selected EB
A2C(0.8)	$1e-3$	0.1
COCOA-reward	$5e-5$	0.01
State-HCA+	$5e-5$	0.01
R-HCA	$5e-5$	0.01

Table 7: Selected hyperparameters for the hindsight model in CartPole.

Algorithm	Selected LRH	Selected K
COCOA-reward	$5e-5$	1
State-HCA+	$5e-5$	1
R-HCA	$5e-5$	1

Table 8: Selected hyperparameters for the actor and critic in Lunar Lander.

Algorithm	Selected LR	Selected EB
A2C(0.8)	$1e-3$	0.1
COCOA-reward	$1e-5$	0.1
R-HCA	$1e-5$	0.1

Table 9: Selected hyperparameters for the hindsight model in Lunar Lander.

Algorithm	Selected LRH	Selected K
COCOA-reward	$1e-4$	2
R-HCA	$1e-4$	2

G R-HCA Algorithm

The R-HCA algorithm is presented in Algorithm 1.

Algorithm 1 R-HCA (Episodic, $\gamma = 1$)

for episode=$1, 2 \dots$ **do**
  $G(i, a) = 0 \quad \forall i, a$
  Collect episode $\tau = (S_0, A_0, R_0, S_1, A_1, R_1, \dots S_H, A_H, R_H, S_{terminal})$ using π_θ
  for $i = 0, 1, 2 \dots H$ **do**
    for $j = i, i+1 \dots H$ **do**
      for $a = 1, 2, \dots |\mathcal{A}|$ **do**
        $G(i, a) = G(i, a) + \frac{\hat{h}^\pi_{j-i}(a|S_i, R_j)}{\pi_\theta(a|S_i)} R_j$
      end for
    end for
  end for
  {Update the actor/policy}
  Take gradient step with loss $L_\pi = -\sum_{i=0}^{H} \sum_{a \in \mathcal{A}} \pi_\theta(a|S_i) G(i, a)$
  {Train the hindsight model}
  Take gradient step with loss $L_h = \sum_{i=0}^{H} \sum_{j=i}^{H} CrossEntropy(\mathbb{I}(A_i), \hat{h}^\pi_{j-i}(.|S_i, R_j))$
end for

H R-HCA with rewarding features

COCOA (Meulemans et al., 2023) introduces COCOA-reward, where $U_k = R_k$, and COCOA-features, where $U_k = f(S_k, A_k, R_k)$ is a reward-encoding feature. Our paper focuses on COCOA-reward, but Proposition 1 generalizes to rewarding features.

Proposition 5. *Consider an action a and state s_t for which $\pi(a|s_t) > 0$, and let $U_{t+k} = f(S_{t+k}, A_{t+k}, R_{t+k})$ be a feature such that R_{t+k} is measurable with respect to (s_t, U_{t+k}), i.e., there exists a function g such that $R_{t+k} = g(s_t, U_{t+k})$. Then*

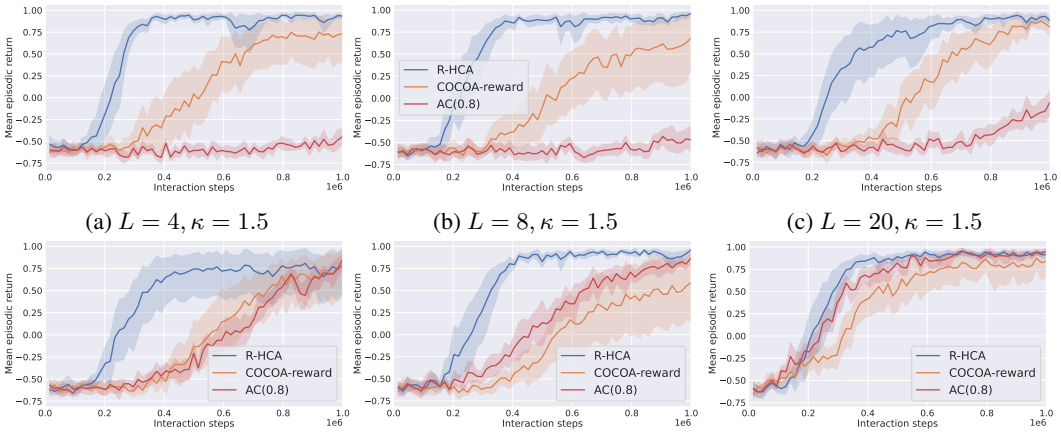
$$Q_\pi(s_t, a) = \mathbb{E}_{\tau \sim \mathcal{T}(s_t, \pi)} \left[\sum_{k \geq 0} \gamma^k \frac{h^\pi_k(a|s_t, U_{t+k})}{\pi(a|s_t)} R_{t+k} \right].$$

Proof. Expanding $Q_\pi(s_t, a) = \sum_u \sum_{k \geq 0} \gamma^k P(U_{t+k} = u|s_t, a)\, g(s_t, u)$ and applying Bayes' rule to replace $P(U_{t+k} = u|s_t, a)$ with $\frac{P(a|s_t, U_{t+k}=u)\, P(U_{t+k}=u|s_t)}{P(a|s_t)}$, we identify the numerator's first factor as the hindsight distribution $h^\pi_k(a|s_t, u)$ and the denominator as $\pi(a|s_t)$. The measurability condition ensures that $g(s_t, u)$ does not depend on a, so converting the sum over u back to an expectation yields the result. □

This gives rise to a feature-based R-HCA update:

$$\hat{\nabla} V_\pi(S_0) = \sum_{t \geq 0} \sum_{a \in \mathcal{A}} \nabla_\theta \pi(a|S_t) \sum_{k \geq 0} \left(\frac{\hat{h}^\pi_k(a|S_t, U_{t+k})}{\pi_\theta(a|S_t)} \right) \gamma^k R_{t+k}.$$

The measurability condition $R_{t+k} = g(s_t, U_{t+k})$ is satisfied by construction in COCOA-features, since the feature function f is defined to encode the reward. COCOA-reward is the special case where $U_k = R_k$ and g is the identity. The variance results from Proposition 2 carry over to this setting with U_{t+k} replacing R_{t+k} in the hindsight model, since the proof depends only on the all-action summation structure of the R-HCA estimator.



(a) $L = 4, \kappa = 1.5$ (b) $L = 8, \kappa = 1.5$ (c) $L = 20, \kappa = 1.5$

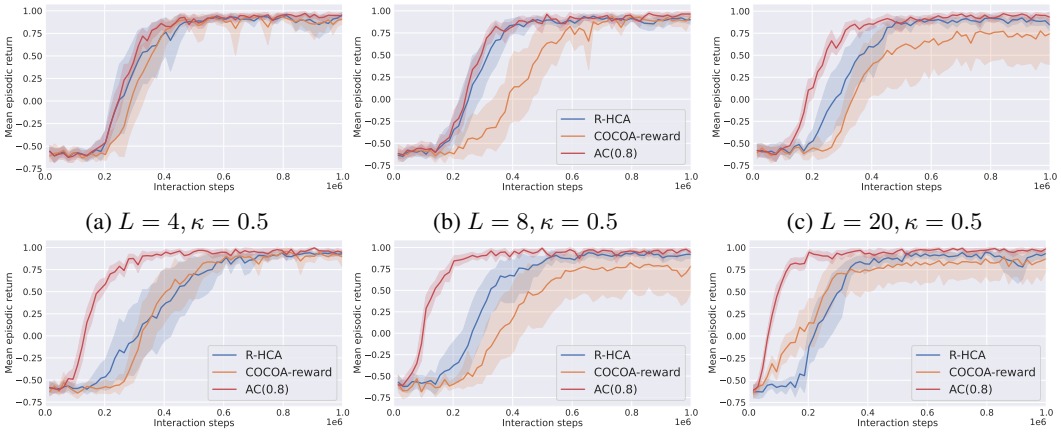
(d) $L = 32, \kappa = 1.5$ (e) $L = 36, \kappa = 1.5$ (f) $L = 40, \kappa = 1.5$

Figure 9: Mean episodic return in the evaluation version of the Planned Catch environment (no noisy rewards). We present results for lock-in $L \in \{4, 8, 20, 32, 36, 40\}$ and $\kappa = 1.5$. The solid line is the average over 10 seeds, and the shading denotes 95% bootstrap confidence intervals.

I Complete results in Planned Catch

Figure 9 presents results for all values of L with $\kappa = 1.5$.

Figure 10 presents results for all values of L with $\kappa = 0.5$.



(a) $L = 4, \kappa = 0.5$ (b) $L = 8, \kappa = 0.5$ (c) $L = 20, \kappa = 0.5$

(d) $L = 32, \kappa = 0.5$ (e) $L = 36, \kappa = 0.5$ (f) $L = 40, \kappa = 0.5$

Figure 10: Mean episodic return in the evaluation version of the Planned Catch environment (no noisy rewards). We present results for lock-in $L \in \{4, 8, 20, 32, 36, 40\}$ and $\kappa = 0.5$. The solid line is the average over 10 seeds, and the shading denotes 95% bootstrap confidence intervals.