RETRACING THE λ-DISCREPANCY: MEMORY LEARNING FROM OFF-POLICY DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Acting optimally under partial observability requires memory. Allen et al. (2024) recently proposed to use the λ-discrepancy, defined as the gap between value estimates obtained at different bootstrapping horizons, as a model-free signal for learning memories that disambiguate latent states. However, the on-policy formulation limits its applicability when fresh interaction data is expensive.

Extending the λ-discrepancy to off-policy learning is nontrivial. When trajectories are generated by a different policy, the resulting multi-step returns no longer correspond to those of the target policy. Importance sampling can correct this mismatch, but often at the cost of prohibitively high variance.

We overcome these issues by introducing the *Retrace discrepancy*, an off-policy generalisation of the λ-discrepancy built from Retrace-style trace corrections (Munos et al., 2016). We first show that the corresponding Retrace operator admits a unique fixed point even without assuming that the observations are Markovian. We then prove that the resulting discrepancy preserves the key partial-observability detection property of the original λ-discrepancy: if a POMDP has a nonzero on-policy λ-discrepancy for some policy, then, for almost every combination of behaviour and target policy in the off-policy setting, the Retrace discrepancy is nonzero as well.

Experiments demonstrate partial-observability detection and memory learning from replay. On Battleship and Minesweeper, the replay-trained agent first exceeds selected mean-return thresholds with $4.4\times$ and $3.0\times$ fewer environment interactions, respectively, than PPO with the on-policy λ-discrepancy.

1 INTRODUCTION

Reinforcement learning agents deployed in the real world rarely have access to the full environment state. Instead, they receive partial, noisy observations from which the relevant latent information must be inferred. Acting near-optimally in such Partially Observable Markov Decision Processes (POMDPs) generally requires an agent to maintain memory of its history: a compact summary of past observations and actions that, together with the current observation, renders the environment approximately Markovian (Puterman, 1994). Recurrent architectures such as RNNs (Amari, 1972), LSTMs (Hochreiter & Schmidhuber, 1997), and GRUs (Cho et al., 2014) provide a natural parametrisation for this memory, but their training relies on the reinforcement signal alone, which does not directly indicate whether the recurrent state is capturing the latent information that matters for long-horizon decisions. A guiding mechanism for the memory quality is therefore necessary to efficiently approximate Markovian settings.

Allen et al. (2024) recently proposed the λ-*discrepancy* as an auxiliary objective for memory learning. Under partial observability, the action-value function Q_π^λ can depend on the bootstrapping horizon λ: different trace lengths expose different consequences of aliasing latent states. A nonzero disagreement $\Lambda_\pi^{\lambda_1,\lambda_2}(\omega,a) = |Q_\pi^{\lambda_1}(\omega,a) - Q_\pi^{\lambda_2}(\omega,a)|$ therefore witnesses missing information relevant to value prediction. Evaluated on memory-augmented observations (ω, m), this supplies a policy- and reward-dependent auxiliary signal for training the memory $\nu : \Omega \times \mathcal{A} \times \mathcal{M} \to \mathcal{M}$. Allen et al. (2024) demonstrate that using this signal alongside the reinforcement-learning objective can improve memory learning, without requiring supervision of the latent state.

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.

A significant practical limitation, however, is that the λ-discrepancy objective is formulated under on-policy data collection. Computing Q_π^λ requires multi-step returns rolled out under the current target policy π, and hence the original approach must generate fresh trajectories after every gradient update. This is prohibitive in settings where environment interaction is expensive, in offline reinforcement learning where only a fixed dataset is available, and in distributed actor-learner systems (Espeholt et al., 2018; Kapturowski et al., 2019) where the actors lag behind the learner and the replay buffer accumulates data from earlier policy versions. The obvious remedy of reusing off-policy transitions from a replay buffer is not straightforward. The n-step returns required by the multi-step estimator are biased when evaluated under a behaviour policy $\mu \neq \pi$, with the bias compounding over the entire look-ahead window. Restoring unbiasedness via importance-sampling corrections (Geist & Scherrer, 2014) is theoretically correct but practically damaging: the product of per-step density ratios can grow exponentially with the horizon, inflating gradient variance to the point where the discrepancy signal is drowned out (Mahmood & Sutton, 2015; Precup et al., 2000).

We address this challenge by introducing the *Retrace discrepancy*, which compares two value functions $Q_{\mu,\pi}^{\lambda_1}$ and $Q_{\mu,\pi}^{\lambda_2}$ for the same target policy π and behaviour policy μ, using different trace parameters. Both are fixed points of the Retrace operator (Munos et al., 2016), whose clipped importance ratios weight future TD residuals within the multi-step update. We first establish that this operator is a contraction even when observations are not Markov, so both value functions are well-defined under partial observability. We then show that, whenever some policy has a nonzero on-policy λ-discrepancy, the Retrace discrepancy is nonzero for almost every behaviour policy and, for each such behaviour, almost every target policy. This preserves the original discrepancy's detection property in the off-policy setting. In our experiments, we investigate using the disagreement between critics estimating these Retrace values as an auxiliary objective for learning memory from replay.

We illustrate the detection signal in T-maze and evaluate memory learning on partially observable Battleship and Minesweeper. In Battleship, Retrace enhanced actor–critic (AC+Retrace) reaches a higher peak mean return than recurrent AC, attaining the model-based Bayes-density reference level. First exceeding mean return eight requires $4.4\times$ fewer interactions than PPO with the on-policy λ-discrepancy. A Minesweeper comparison extends these results: AC+Retrace first exceeds mean return zero with $3.0\times$ fewer interactions while speeding up training performance compared to recurrent AC.

Our main contributions are:

- We define the Retrace discrepancy and establish generic off-policy detection whenever an on-policy detection witness with λ-discrepancy bigger than zero exists, while retaining zero discrepancy for Markov-sufficient observations.
- We show that off-policy evaluation can detect partial observability beyond the on-policy discrepancy, using Parity Check as an explicit separation.
- We integrate the Retrace discrepancy into a recurrent AC algorithm and demonstrate memory-learning and sample-efficiency benefits on Battleship and Minesweeper.

2 Background

Following the notation of Allen et al. (2024), an MDP (Puterman, 1994) is a tuple $(\mathcal{S}, \mathcal{A}, R, T, \gamma)$ with state and action spaces $\mathcal{S}$ and $\mathcal{A}$, reward function $R(s,a)$, transition kernel $T(s' \mid s,a)$, and discount $\gamma \in [0,1)$. Taking action a_t in state s_t yields reward r_{t+1} and next state s_{t+1}. For a policy $\pi : \mathcal{S} \to \Delta\mathcal{A}$, define the return $g_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i+1}$ and values $V_\pi(s) = \mathbb{E}_\pi[g_t \mid s_t = s]$ and $Q_\pi(s,a) = \mathbb{E}_\pi[g_t \mid s_t = s, a_t = a]$. These are fixed points of the corresponding Bellman operators.

Sutton (1988) introduced the TD(λ) estimator, a geometric mixture of n-step returns. For a candidate value function V, define

$$
\begin{aligned}
&g_{t:t+n}(V) = \textstyle\sum_{i=0}^{n-1} \gamma^i r_{t+i+1} + \gamma^n V(s_{t+n}),\\
&(\mathcal{T}_\pi^\lambda V)(s) = \mathbb{E}_\pi\left[(1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} g_{t:t+n}(V) \,\middle|\, s_t = s\right].
\end{aligned}
\tag{1}
$$

for $\lambda < 1$, with $\lambda = 1$ denoting the Monte Carlo limit. We write V_π^λ for the fixed point of this operator and define Q_π^λ analogously. In an MDP, $V_\pi^\lambda = V_\pi$ and $Q_\pi^\lambda = Q_\pi$ for every λ.

2.1 Partial Observability and the λ-Discrepancy

In the POMDP setting we additionally have an observation function $\Phi : \mathcal{S} \to \Delta\Omega$, with Ω a set of observations, describing the probability of seeing observation ω in latent state s. The agent receives ω_t rather than s_t and must act on observations alone.

We can connect back to the MDP setting by treating observations as pseudo-states with adapted transition and reward functions $T_\Omega(\omega' \mid \omega, a) = \sum_{s,s' \in \mathcal{S}} \Phi(\omega' \mid s') T(s' \mid s, a) \Pr(s \mid \omega)$ and $R_\Omega(\omega, a) = \sum_{s \in \mathcal{S}} R(s, a) \Pr(s \mid \omega)$, where $\Pr(s \mid \omega)$ is the policy-dependent posterior over latent states. This is the effective MDP model of Allen et al. (2024). Although this model captures one-step conditional quantities, it need not reproduce the multi-step behaviour of the actual observation process.

The λ-discrepancy (Allen et al., 2024) exploits this mismatch. When TD(λ) returns are formed from actual POMDP trajectories and value functions depend only on observations, their fixed points Q_π^λ can depend on λ: different trace lengths balance observed rewards against bootstrapping from an observation that may alias latent states. This dependence arises from the non-Markov observation process; returns generated by the effective MDP itself would retain the MDP invariance described above. Allen et al. (2024) define

$$\Lambda_\pi^{\lambda_1, \lambda_2}(\omega, a) \;=\; \left| Q_\pi^{\lambda_1}(\omega, a) - Q_\pi^{\lambda_2}(\omega, a) \right|. \tag{2}$$

When the observation and action arguments are omitted, we write $\Lambda_\pi^{\lambda_1, \lambda_2} = \|\Lambda_\pi^{\lambda_1, \lambda_2}(\cdot, \cdot)\|$ for the norm of the vector of pointwise discrepancies over the evaluated observation–action pairs. We use a fixed norm that includes every evaluated coordinate; the zero/nonzero distinction is independent of this choice. A nonzero discrepancy witnesses that the current representation is not Markov for the evaluated policy and reward function. The converse does not hold in general: a zero discrepancy need not imply full observability or a sufficient representation.

Beyond diagnosing partial observability, equation 2 can be leveraged as a training signal for the memory needed to address it. Define a memory function $\nu : \Omega \times \mathcal{A} \times \mathcal{M} \to \mathcal{M}$ updating an internal memory state $m_t = \nu(\omega_t, a_{t-1}, m_{t-1})$, and write $\tilde{\omega}_t = (\omega_t, m_t) \in \Omega_\mathcal{M}$ for the resulting augmented observation, where $\Omega_\mathcal{M} = \Omega \times \mathcal{M}$. The corresponding memory-augmented POMDP has latent states $\mathcal{S}_\mathcal{M} = \mathcal{S} \times \mathcal{M}$ and the original action space $\mathcal{A}$; memory is updated internally by ν rather than chosen by the agent.

Driving equation 2 towards zero on the augmented observation $\tilde{\omega} = (\omega, m)$ encourages the memory to retain latent information relevant to long-horizon value prediction. The memory function ν may be implemented as a Recurrent Neural Network (Amari, 1972), a Long Short-Term Memory network (Hochreiter & Schmidhuber, 1997), or a Gated Recurrent Unit (Cho et al., 2014).

2.2 Off-Policy Learning in Return-Based RL

In off-policy learning we estimate values for a target policy π from data collected under a behaviour policy μ. Multi-step return estimates built from μ-data are biased estimators of on-policy quantities, and the bias compounds with the look-ahead horizon. The same issue arises in distributed actor–learner setups where the actors lag behind the learner policy (Espeholt et al., 2018). It directly affects our setting because the λ-discrepancy requires multi-step returns evaluated under π.

Munos et al. (2016) introduce a general operator that subsumes the standard return-based off-policy algorithms,

$$\begin{aligned} \mathcal{R}Q(\omega, a) &= Q(\omega, a) + \mathbb{E}_\mu \left[\sum_{t \geq 0} \gamma^t \Big(\prod_{u=1}^{t} c_u \Big) \delta_t^\pi(Q) \;\Big|\; \omega_0 = \omega, a_0 = a \right], \\ \delta_t^\pi(Q) &= r_{t+1} + \gamma\, \mathbb{E}_{b \sim \pi(\cdot \mid \omega_{t+1})} Q(\omega_{t+1}, b) - Q(\omega_t, a_t). \end{aligned} \tag{3}$$

parameterised by a per-step trace coefficient c_u. Importance sampling (Geist & Scherrer, 2014) sets $c_u = \pi(a_u \mid \omega_u)/\mu(a_u \mid \omega_u)$, which is unbiased but suffers from severe variance, even with reduction techniques (Mahmood & Sutton, 2015). Tree-backup (Precup et al., 2000) uses $c_u = \lambda\, \pi(a_u \mid \omega_u)$, avoiding the ratio at the cost of inefficiency in near-on-policy regimes. Retrace(λ) (Munos et al., 2016) sets $c_u = \lambda \min\big(1, \pi(a_u \mid \omega_u)/\mu(a_u \mid \omega_u)\big)$, truncating large ratios while retaining long traces when the policies are similar.

3 METHOD

Our goal is to generalise the λ-discrepancy to detect partial observability from off-policy data by comparing value functions with different trace lengths. Retrace provides a natural starting point, but its usual fixed-point guarantee assumes an MDP. We therefore first establish that Retrace defines a unique value function on observations even when their dynamics are not Markov. We then use the disagreement between two such value functions to define the *Retrace discrepancy*, and show that it preserves the original λ-discrepancy's partial-observability detection property.

Setting. For the theoretical analysis, we adopt a finite, discounted setting. We fix a POMDP with observations $\omega \in \Omega$ as in Section 2.1, finite latent-state, observation, and action spaces, bounded rewards, and $0 < \gamma < 1$. We additionally assume at least two actions. Both policies act on the current observation ω, with policy space $\Pi = \prod_{\omega \in \Omega} \Delta(\mathcal{A})$. We do not assume that ω_t is Markov. A complete description of the finite-model construction is provided in Appendix A.1.

3.1 A WELL-DEFINED RETRACE VALUE UNDER PARTIAL OBSERVABILITY

We specialise the return-based operator in equation 3 to the standard Retrace coefficients,

$$c^{\lambda}_{\mu,\pi}(\omega, a) = \lambda \min\left\{1, \frac{\pi(a \mid \omega)}{\mu(a \mid \omega)}\right\}, \qquad \lambda \in [0, 1], \tag{4}$$

with $c^{\lambda}_{\mu,\pi} = 0$ when $\mu(a \mid \omega) = 0$. Let $\delta^{\pi}_t(q)$ denote the target-policy TD residual in equation 3. The resulting operator is

$$\begin{aligned}(\mathcal{R}^{\lambda}_{\mu,\pi} q)(\omega, a) &= q(\omega, a) \\ &\quad + \mathbb{E}_{\mu}\left[\sum_{t=0}^{\infty} \gamma^t \left(\prod_{u=1}^{t} c^{\lambda}_{\mu,\pi}(\omega_u, a_u)\right) \delta^{\pi}_t(q) \,\middle|\, \omega_0 = \omega, a_0 = a\right].\end{aligned} \tag{5}$$

The expectation averages over behaviour-policy trajectories conditional on the starting observation ω and action a. We use the sampling convention of Allen et al. (2024), detailed in Appendix A.3, and keep the starting distribution fixed when comparing target policies or trace parameters. Throughout, two-policy subscripts list the behaviour policy μ first and the target policy π second.

The empty product at $t = 0$ is one. Thus $\lambda = 0$ retains only the one-step update, whereas larger λ allows later residuals to contribute. Policy mismatch can shorten these traces through the clipped ratios. In particular, $\lambda = 1$ need not give a Monte Carlo return off policy.

The question is whether repeatedly applying this operator has a well-defined limit even when observations do not resolve the latent state. The original Retrace analysis cannot be applied directly, as a Bellman equation on ω need not describe the actual multi-step observation process. Our first result establishes contraction without that assumption.

Theorem 1 (Contraction under partial observability)**.** *For fixed policies $\mu, \pi \in \Pi$ and a fixed starting-state sampling law, the operator $\mathcal{R}^{\lambda}_{\mu,\pi}$ in equation 5 is a γ-contraction in the supremum norm and has a unique fixed point. More generally, the same conclusion holds when the trace coefficient is replaced by any fixed nonnegative function c satisfying*

$$0 \le \mu(a \mid \omega) c(\omega, a) \le \pi(a \mid \omega) \quad \textit{for every } (\omega, a).$$

The proof lifts value functions to a latent Markov process, applies the trace-weighted update there, and averages back over states sharing the same observation. Trace domination makes the resulting linear weights nonnegative with row sums at most γ, averaging over aliased states preserves this bound. Appendix A.2 gives the full argument. Consequently, each choice of λ defines an unambiguous observation-based value function under partial observability.

3.2 FROM RETRACE VALUES TO A DISCREPANCY

For distinct $\lambda_1, \lambda_2 \in [0, 1]$, define

$$\begin{aligned} Q^{\lambda}_{\mu,\pi} &= \mathrm{Fix}(\mathcal{R}^{\lambda}_{\mu,\pi}), \\ D^{\lambda_1,\lambda_2}_{\mu,\pi}(\omega, a) &= \left|Q^{\lambda_1}_{\mu,\pi}(\omega, a) - Q^{\lambda_2}_{\mu,\pi}(\omega, a)\right|, \\ D^{\lambda_1,\lambda_2}_{\mu,\pi} &= \left\|D^{\lambda_1,\lambda_2}_{\mu,\pi}(\cdot, \cdot)\right\|, \end{aligned} \tag{6}$$

where $\mathrm{Fix}(\cdot)$ denotes the fixed point of an operator, $\|\cdot\|$ can be any norm. We call $D^{\lambda_1,\lambda_2}_{\mu,\pi}(\omega, a)$ the pointwise *Retrace discrepancy* and $D^{\lambda_1,\lambda_2}_{\mu,\pi}$ its aggregate. The idea is the same as for the λ-discrepancy: both Q-functions evaluate the same target policy with different trace parameters, estimated from the same behaviour policy. Under partial observability, the resulting fixed points can disagree because different trace lengths expose different consequences of aliasing latent states.

Two properties make this construction a generalisation of the original λ-discrepancy. First, the on-policy case directly reduces to it, i.e. when using the same behaviour and target policy, $\mu = \pi$, every trace coefficient $c^{\lambda}_{\pi,\pi}$ equals λ. Equation 5 then reduces to the TD(λ) return operator, giving

$$D^{\lambda_1,\lambda_2}_{\pi,\pi}(\omega, a) = \Lambda^{\lambda_1,\lambda_2}_{\pi}(\omega, a) \tag{7}$$

on all reachable observation–action pairs, and hence $D^{\lambda_1,\lambda_2}_{\pi,\pi} = \Lambda^{\lambda_1,\lambda_2}_{\pi}$ when aggregated over those pairs with the same norm.

Second, when observations are sufficient to define an MDP, the target-policy value satisfies the Bellman equation independently of trace length. It is consequently a fixed point of every Retrace operator, and uniqueness gives the following result.

Corollary 2 (Vanishing for Markov-sufficient observations)**.** *Suppose ω is a Markov-sufficient observation of the controlled process, so that its reward and observation-transition laws depend only on (ω, a). Then $Q^{\lambda}_{\mu,\pi} = Q_\pi$ for every behaviour/target pair and every $\lambda \in [0, 1]$. Hence $D^{\lambda_1,\lambda_2}_{\mu,\pi}(\omega, a) = 0$ at every evaluated pair, and $D^{\lambda_1,\lambda_2}_{\mu,\pi} = 0$ for all λ_1, λ_2.*

A nonzero Retrace discrepancy thus witnesses partial observability relevant to value prediction. As with the on-policy discrepancy, zero disagreement alone does not establish that observations are sufficient: the chosen rewards and policies may fail to reveal the missing information.

3.3 Preserving the partial-observability signal off-policy

The remaining question is whether off-policy trace clipping can erase a signal that the on-policy discrepancy would detect. Our main detection result shows that, if a single policy exposes a nonzero on-policy discrepancy, the Retrace discrepancy is nonzero for almost every behaviour policy and, for each such behaviour, almost every target policy. Here, "almost every" refers to Lebesgue measure in the $d = |\Omega|(|\mathcal{A}| - 1)$ independent coordinates of the policy space, after discarding observations unreachable under any policy.

Theorem 3 (Generic off-policy detection)**.** *Fix distinct $\lambda_1, \lambda_2 \in [0, 1]$ and use the sampling convention of Appendix A.3 with a common initial distribution. Suppose there exists $\beta_0 \in \Pi$ whose supported on-policy discrepancy satisfies $\Lambda^{\lambda_1,\lambda_2}_{\beta_0} > 0$. Then there is a Lebesgue-null set $E \subset \Pi$, containing the relative boundary of Π, such that, for every $\mu \in \Pi \setminus E$,*

$$\mathrm{Leb}_d\{\pi \in \Pi : D^{\lambda_1,\lambda_2}_{\mu,\pi} = 0\} = 0.$$

The key idea is to use the on-policy case as a witness for off-policy detection. The on-policy analytic argument of Allen et al. (2024) makes a nonzero discrepancy generic among behaviour policies. For each such behaviour μ, equation 7 supplies a nonzero Retrace discrepancy at $\pi = \mu$. Clipping prevents global analyticity in the target policy, but partitions the policy space into finitely many regions within which the fixed points are analytic. Each region approaches $\pi = \mu$, so continuity supplies a nonzero witness in every region, the analytic zero-set argument then applies region by region. Finally, a witness using only a subset of actions survives a sufficiently small enlargement of its support. See appendices A.4 and A.5 for the full proof.

Together, the results establish Retrace disagreement as an off-policy diagnostic of partial observability: it vanishes for Markov-sufficient observations, while a signal detectable by the original discrepancy persists for generic policy pairs. Interestingly, the Retrace discrepancy allows us to detect partial observability even in cases where the original λ-discrepancy cannot. We highlight this in the remainder of this section.

3.4 Detection beyond the on-policy discrepancy

In the Parity Check environment of Allen et al. (2024), an agent observes independent uniform red/blue and pink/cyan colors, then a white junction (Figure 1a). UP gives $+1$ when the color families

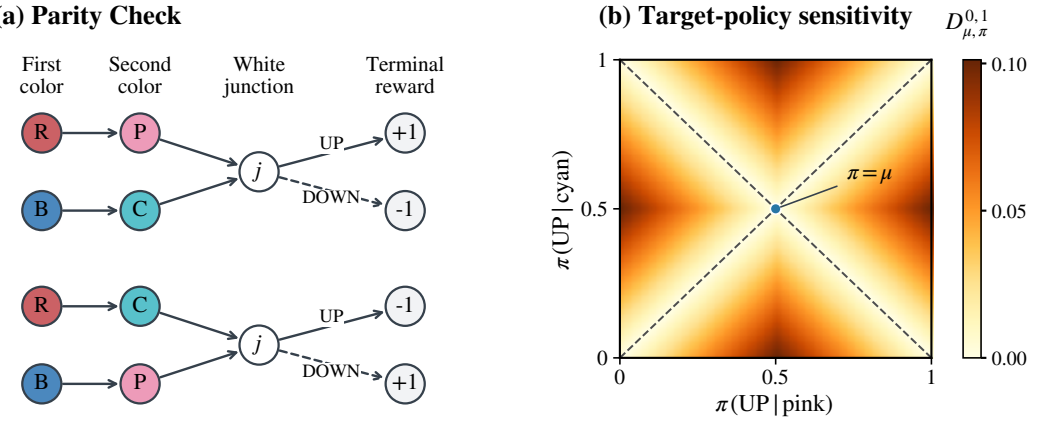

Figure 1: **Detection beyond the on-policy discrepancy.** (a) Parity Check, redrawn from Allen et al. (2024). Each initial path has probability $1/4$. R, B, P, and C denote red, blue, pink, and cyan. (b) Exact Retrace discrepancy $D^{0,1}_{\mu,\pi}$ at $\gamma = 0.9$ as the target's UP probabilities at pink and cyan vary. The behaviour selects UP with probability $3/4$ at white and $1/2$ elsewhere. The target matches it except at the two plotted coordinates. Dashed lines mark the zero set, the dot marks the on-policy point $\pi = \mu$.

match (red–pink or blue–cyan) and -1 otherwise. DOWN reverses these rewards. Earlier actions only advance the episode with zero reward, the junction decision terminates it. Every memoryless policy has zero on-policy λ-discrepancy.

Choose both policies to select UP with probability $1/2$, except that both use $3/4$ at white and the target additionally uses $3/4$ at cyan. Clipping weights the pink and cyan continuations differently, breaking the on-policy reward cancellation. With $\alpha = 0$, the supremum norm, $0 < \gamma < 1$, and distinct $\lambda_1, \lambda_2 \in [0, 1]$, this gives

$$D^{\lambda_1,\lambda_2}_{\mu,\pi} = \|Q^{\lambda_1}_{\mu,\pi} - Q^{\lambda_2}_{\mu,\pi}\|_\infty = \frac{\gamma^2}{16}|\lambda_1^2 - \lambda_2^2| > 0. \tag{8}$$

This strictly enlarges the class of environments admitting a detection witness. Moreover, every fully supported behaviour policy detects Parity Check for almost every target policy. Figure 1b illustrates one such policy slice. Appendix B gives the formal setup, worked calculation, and general proof.

4 Experiments

We ask whether the Retrace discrepancy can guide useful memory learning from replay and whether reusing off-policy experience improves sample efficiency.

From discrepancy to memory learning. Following Allen et al. (2024, Figure 3), we illustrate the effect of missing information in a five-cell T-maze. An initial cue identifies the rewarding branch, but aliased observations can hide the goal direction and the agent's position later in the episode. We interpolate between fully observed and aliased observation channels,

$$\Phi_\eta = (1-\eta)\Phi_{\text{perfect}} + \eta\Phi_{\text{aliased}}, \qquad \eta \in [0, 1], \tag{9}$$

where Φ_{aliased} merges corridor states, junction states, or both, while preserving the initial cue. For $\eta = 0$, the environment is therefore fully observable. Figure 2 compares exact discrepancies along these paths. The discrepancy vanishes under full observability and reveals missing information both on policy and under policy mismatch. With junction-only aliasing, the balanced behaviour has zero on-policy discrepancy, yet off-policy evaluation reveals the biased target's signal.

Finite-data detection. With fixed policies, neural critics recover the signal from off-policy replay. At 28,672 transitions, estimated RMS discrepancy is 0.158 ± 0.010 under full combined aliasing

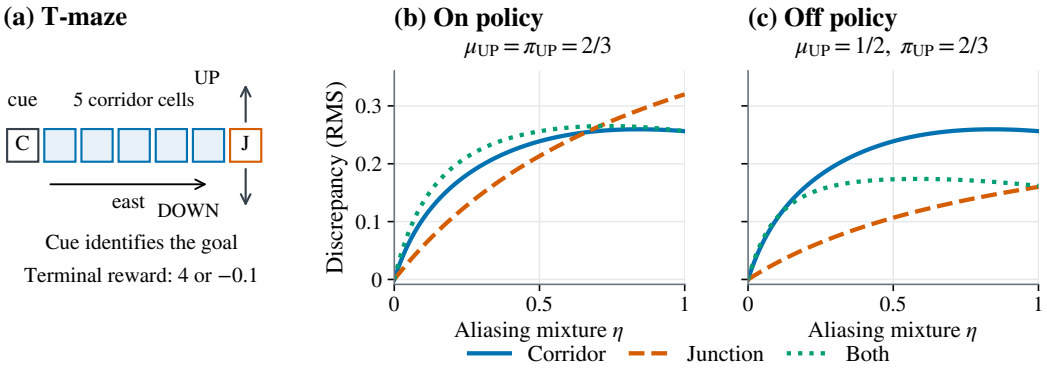

Figure 2: **Missing information produces a discrepancy signal.** (a) Five-cell T-maze. (b) On-policy λ-discrepancy with junction-UP probability $2/3$. (c) The same target with behaviour junction-UP probability $1/2$. Both policies move east before the junction. Curves show exact fixed-point RMS discrepancies at $\gamma = 0.9$ and $(\lambda_1, \lambda_2) = (0, 1)$. Details are in Appendix C.1.

(exact: 0.162), compared with $(1.5 \pm 1.0) \times 10^{-4}$ under full observability (mean $\pm$ SD over five seeds). Appendix C.2 reports the data-budget sweep and separates sampling from fitting error.

We use critic disagreement to train recurrent memory alongside the control objective, extending Allen et al. (2024) to replay. Its utility remains an empirical question because discrepancy need not vary monotonically with information.

Recurrent replay setup. Following R2D2 (Kapturowski et al., 2019), we sample contiguous replay windows with B burn-in steps and L learning steps. Burn-in reconstructs the recurrent state under current parameters without gradients; learning uses truncated backpropagation through time. Carries reset at episode boundaries. Retrace targets correct policy mismatch using the current target policy and recorded behaviour probabilities.

Memory architecture and auxiliary objective. All four heads in Figure 3 read the recurrent state $m_t = \nu_\theta(\omega_t, a_{t-1}, m_{t-1})$ directly. The actor $\pi_\varphi(\cdot \mid m_t)$ and task critic $Q_\phi(m_t, \cdot)$ optimise control. The auxiliary heads $q^{\lambda_1}_{\psi_1}(m_t, \cdot)$ and $q^{\lambda_2}_{\psi_2}(m_t, \cdot)$ evaluate the same target policy at distinct trace parameters, with their disagreement guiding memory learning.

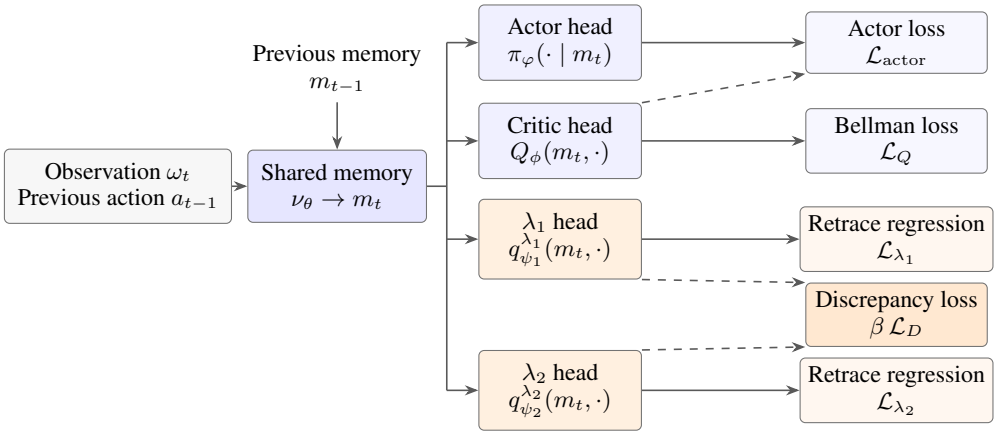

Figure 3: **Recurrent architecture and training losses.** Each value head represents a twin pair. Dashed connections freeze value-head parameters; the discrepancy gradient trains memory. The losses combine as in equation 10.

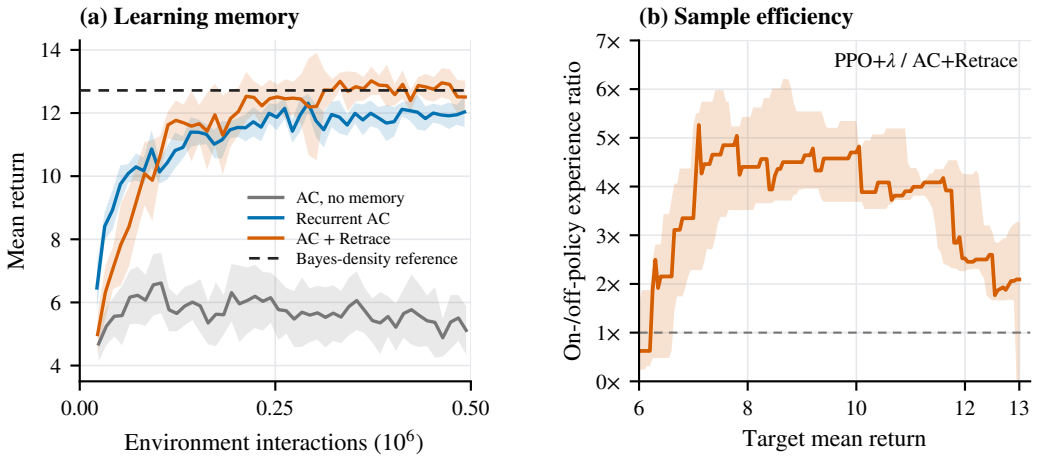

Figure 4: **Memory learning and sample efficiency in Battleship.** AC+Retrace attains the Bayes-density reference level and a higher peak mean return than recurrent AC. It first exceeds mean return eight with $4.4\times$ fewer interactions than PPO+λ. (a) Mean evaluation return through $500{,}000$ interactions with pointwise 95% CIs over ten seeds. The Bayes-density line is an exact expectation. (b) PPO+λ interactions divided by AC+Retrace interactions (y-axis) to first exceed each target mean return (x-axis); shading gives a 95% seed-bootstrap interval. Interaction counts include random replay prefill.

Training objective. We use a standard entropy-regularised actor loss $\mathcal{L}_{\text{actor}}$ and mean-squared regression of the one-step TD residual in equation 3 for $\mathcal{L}_Q$. The losses $\mathcal{L}_{\lambda_1}$ and $\mathcal{L}_{\lambda_2}$ fit truncated targets from the Retrace operator equation 5 by mean-squared regression. The memory loss $\mathcal{L}_D$ is the replay-averaged squared pointwise discrepancy in equation 6, estimated by the two λ heads on m_t. Targets are detached; value-head parameters are held fixed in the actor and discrepancy terms. In the recurrent experiments below, actor gradients also stop at the memory, critic and discrepancy losses train it. The joint objective is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{actor}} + \mathcal{L}_Q + \mathcal{L}_{\lambda_1} + \mathcal{L}_{\lambda_2} + \beta\,\mathcal{L}_D, \qquad \beta \geq 0, \tag{10}$$

where β weights the memory loss. Appendix G gives the full objectives.

4.1 Learning useful memory from replay

We evaluate on 5×5 Battleship with ships of lengths three and two. The agent observes the latest shot outcome and must remember previous outcomes to locate the remaining ship cells. Reward is -1 at each step and 25 when all ships are hit. Actions which were already taken are masked out, so termination is guaranteed after 25 steps, making the undiscounted return $26 - N$ for clearing the board in N shots. Figure 4a compares memoryless AC, recurrent AC, and recurrent AC with Retrace discrepancy over ten seeds and up to $500{,}000$ environment interactions. The dashed line is a Bayes-density player supplied with the full shot record and placement model (Appendix D.1).

The discrepancy introduces a slower startup, but subsequently reaches the model-based reference level and a higher peak mean return than recurrent AC: 13.02 ± 0.45 versus 12.31 ± 0.29 (95% CIs). Retuned PPO and PPO+λ reach peak means of 12.16 ± 0.62 and 13.48 ± 0.24, respectively (Appendix D.2). Thus the Retrace discrepancy achieves comparable peak performance to the on-policy λ-discrepancy.

4.2 Sample efficiency

For method A, let $T_A(r)$ be the first recorded interaction count at which its mean return exceeds r. Figure 4b plots $T_{\text{PPO}+\lambda}(r)/T_{\text{AC+Retrace}}(r)$ against target return, so values above one favour

replay-based learning. We count all training interactions, including the AC agents' initial random replay fill, and bootstrap whole seed trajectories for uncertainty (Appendix D.3).

The mean first exceeds return eight after $62\,800$ interactions with Retrace discrepancy and $276\,480$ with PPO+λ: a $4.4\times$ advantage (95% interval $[3.5, 5.4]$). The corresponding factors at returns ten and twelve are $4.7\times$ and $2.5\times$. Replay therefore reaches useful performance with substantially less interaction, despite the auxiliary objective's initial delay.

We additionally evaluate on partially observed 6×6 Minesweeper over twenty seeds per method. Minesweeper provides neither an action mask nor a persistent board observation, so the agent must retain earlier clues and queried locations in memory. This tests the auxiliary objective in a setting where memory also helps avoid repeated actions. Both recurrent AC variants substantially outperform the memoryless control. AC+Retrace again speeds up learning and achieves a higher peak mean return than recurrent AC. A mean return of zero is first exceeded after $636\,016$ interactions, compared with $1\,898\,496$ for retuned PPO+λ, a $3.0\times$ experience ratio (95% bootstrap interval $[2.8, 3.4]$). This extends the benefits to a second environment for the selected configurations, see Appendix E for more details.

5 Limitations & Future Work

Retrace discrepancy provides a policy- and reward-dependent signal of missing information. As with the on-policy λ-discrepancy, zero disagreement need not imply Markov sufficiency, and its magnitude need not decrease monotonically as information is added (Appendices B.2 and C.4). We use this signal as an auxiliary objective alongside the control losses, characterising how their joint optimisation builds useful memory is a natural next theoretical step. Such an analysis would extend our current guarantees, which concern exact fixed points in finite discounted POMDPs with fixed representations, policies, and sampling laws. Since generic detection does not bound signal magnitude or sample complexity, quantitative error bounds would also clarify how critic estimation error affects the measured discrepancy as learning progresses. Our experiments evaluate memory learning from replay. Extending this evaluation to longer-horizon tasks and richer observations would assess transfer of these benefits. The ability to reuse experience also motivates learning memory from a buffer populated once, for example from human demonstrations, for use in fully offline control or imitation learning. Our current memory-learning experiments continue to collect experience, evaluating these further applications would involve questions of reward availability, dataset coverage, and trace correction when behaviour probabilities are unrecorded or depend on history beyond the learner's representation.

6 Conclusion

We introduced the Retrace discrepancy as a model-free auxiliary objective for learning memory from off-policy replay. In finite discounted POMDPs, we established a unique observation-based Retrace fixed point and proved that a nonzero on-policy detection witness implies detection for almost every behaviour–target policy pair. The discrepancy remains zero for Markov-sufficient observations. Parity Check further shows that off-policy evaluation can reveal partial observability invisible to the on-policy discrepancy under every memoryless policy.

Our experiments connect this detection signal to memory learning from finite data. Neural critics recover the signal from T-maze replay, and AC+Retrace attains the model-based Bayes-density reference level in Battleship, improving the peak return for recurrent AC and comparable peak performance to retuned PPO+λ. For the evaluated configurations, AC+Retrace first exceeds mean return eight in Battleship and zero in Minesweeper using $4.4\times$ and $3.0\times$ fewer environment interactions than PPO+λ, respectively.

Together, these results show that critic disagreement can guide useful memory learning from replay, reducing the fresh interaction needed to reach shared return targets. Retrace discrepancy thus provides explicit auxiliary supervision for task-relevant memory from reusable experience, without requiring latent-state labels or an environment model.

AI USE STATEMENT

In this work, we used generative AI tools for providing critical ingredients for proving mathematical claims and assisting in the writing of proofs. We have not used generative AI tools for helping to develop theoretical models or conceptual frameworks, formulate mathematical claims, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning or reformatting datasets or interpreting results. Generating synthetic data sets, assisting with translation, cleaning or reformatting datasets and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for creating the figures in this paper from the results, creating and editing software code, editing this paper to improve readability, and proposing a title and keywords for this paper. We have reviewed all AI-assisted work and elaborate on the specifics of usage below. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Regarding Mathematical Results: Under the recommendation of colleagues from the Mathematics department, we used Claude Fable 5 (Anthropic) to help with proofs, and had multiple people verify the output (switching to GPT 5.6 Sol when it became available and checking correctness again with GPT 6 Astra when it became available). The overall machinery and theoretical story are our own, and we note that LLMs, while capable of providing very good proofs, were generally not able to see the big-picture and were often also only able to give proofs after being given a clear sketch or the central idea. One exception is appendix A.4, where LLMs supplied the critical idea of cellwise analyticity. Finally, GPT 6 Astra was used to help deciding how to divide our results between main paper and appendix.

Regarding Writing: We tried generating a draft for the methods and results section of the paper from the git repository. This did not work, often over-emphasising specific implementation details as method and leaving out actual methodological details, so we re-wrote both sections from the ground up. We report this for full transparency and because we cannot absolutely ensure that no idea from this original draft has made it into the main part of the paper. We subsequently used GPT 6 Astra only for shortening sections and editing them to make them more understandable as well as generating figures from our results. We did, however, use GPT 6 Astra to provide a draft for a concise representation of the many notes and sessions of the theoretical work as well as provide explanations from our code repository and experiment platform, which we used as a basis for most of the Appendix.

Regarding Code: The core method was implemented by hand to ensure correctness. We used LLMs (Claude Fable 5, Opus 4.1, 4.6, 4.7, 4.8, 5 and Sonnet 4 from Anthropic and GPT 5.6 Sol, Terra, Luna as well as GPT 6 Astra from OpenAI) for code cleaning and documentation, creating experiment runners and hyperparameter sweeps, as well as assistants for debugging and fast prototyping of solution candidates. All code was carefully reviewed and we made sure that the main methodological parts were off-limit for any LLM edits (except for documentation and removal of prototyped future directions).

ETHICS STATEMENT

This work focuses on methodological questions in reinforcement learning. Our theoretical analysis and experiments in simulated environments involve neither human participants nor personal data, and we identify no direct ethical concerns specific to the study. However, the method is general-purpose and could support both beneficial and harmful applications as dual use cannot be ruled out and downstream ethical implications depend on the deployment context.

REPRODUCIBILITY STATEMENT

The code used for this work can be found at: `https://anonymous.4open.science/r/retrace-discrepancy-6EE4`. The assumptions and proofs supporting our theoretical claims are provided in Appendices A and B. Section 4 describes the agent architecture, training objectives and recurrent replay procedure. Appendices C and D document the experimental environments, selected hyperparameters, evaluation protocols and statistical analyses, including the retuning of the on-policy baselines. The hardware used for the experiments and project development is described in Appendix F.1.

REFERENCES

Cameron Allen, Aaron Kirtland, Ruo Yu Tao, Sam Lobel, Daniel Scott, Nicholas Petrocelli, Omer Gottesman, Ronald Parr, Michael Littman, and George Konidaris. Mitigating partial observability in sequential decision processes via the lambda discrepancy. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 62988–63028. Curran Associates, Inc., 2024. doi: 10.52202/079017-2011. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/73073ccb3bc559fd001e66b9079d6d5e-Paper-Conference.pdf.

S.-I. Amari. Learning patterns and pattern sequences by self-organizing nets of threshold elements. *IEEE Transactions on Computers*, C-21(11):1197–1206, 1972. doi: 10.1109/T-C.1972.223477.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://aclanthology.org/D14-1179/.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1407–1416. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/espeholt18a.html.

Matthieu Geist and Bruno Scherrer. Off-policy learning with eligibility traces: a survey. *J. Mach. Learn. Res.*, 15(1):289–333, January 2014. ISSN 1532-4435.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8): 1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Steven Kapturowski, Georg Ostrovski, John Quan, Rémi Munos, and Will Dabney. Recurrent experience replay in distributed reinforcement learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=r1lyTjAqYX.

A. Rupam Mahmood and Richard S. Sutton. Off-policy learning based on weighted importance sampling with linear computational complexity. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, UAI'15, pp. 552–561, Arlington, Virginia, USA, 2015. AUAI Press. ISBN 9780996643108.

B. S. Mityagin. The zero set of a real analytic function. *Mathematical Notes*, 107(3):529–530, Mar 2020. ISSN 1573-8876. doi: 10.1134/S0001434620030189. URL https://doi.org/10.1134/S0001434620030189.

Remi Munos, Tom Stepleton, Anna Harutyunyan, and Marc Bellemare. Safe and efficient off-policy reinforcement learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/c3992e9a68c5ae12bd18488bc579b30d-Paper.pdf.

Doina Precup, Richard S. Sutton, and Satinder P. Singh. Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pp. 759–766, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1558607072.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 1994. ISBN 978-0-47161977-2. doi: 10.1002/9780470316887. URL https://doi.org/10.1002/9780470316887.

Richard S. Sutton. Learning to predict by the methods of temporal differences. *Mach. Learn.*, 3(1):9–44, August 1988. ISSN 0885-6125. doi: 10.1023/A:1022633531479. URL https://doi.org/10.1023/A:1022633531479.

A Partial Observability Detection by the Retrace Discrepancy

Relation to prior work. Many of the proofs here are extensions from existing ideas in other work. We use the discounted-occupancy posterior of Allen et al. (2024) and adapt their analytic-policy argument (Appendices C, D, and G). The TD-residual operator, clip-at-one traces, trace-domination condition, and positive-weight contraction argument come from Munos et al. (2016, Eq. (3), Theorem 1, and Section 4). Our analysis below supplies the projected finite-model formulation, the clipping-cell/diagonal-witness argument, and the support-enlargement step connecting supported witnesses to generic off-policy detection. The on-policy almost-everywhere conclusion is explicitly attributed to Allen et al. (2024, Theorem 1).

A.1 Finite model, sampling, and the residual operator

Assume finite latent-state, observation, and action spaces, $|\mathcal{A}| \geq 2$, bounded rewards $|r_{t+1}| \leq R_{\max}$, and $0 < \gamma < 1$. Fix the POMDP throughout. To handle stochastic observations while keeping the projection explicit, let $z_t = (s_t, \omega_t)$ be the controlled Markov state on the state–observation space $\mathcal{Z}$, with $\omega_t = g(z_t)$ and transition kernel

$$T_{\mathcal{Z}}((s', \omega') \mid (s, \omega), a) = T(s' \mid s, a)\Phi(\omega' \mid s').$$

More generally, the arguments apply to any finite controlled Markov state z_t with transition kernel $T_{\mathcal{Z}}$ and observation map g. No Markov assumption is made about ω_t itself. Set $r(z, a) = \mathbb{E}[r_{t+1} \mid z_t = z, a_t = a]$. Initially take g to be surjective and $\mathcal{Z}$ closed under all controlled transitions. A fixed initial law ζ and the appropriate reachable restrictions are specified below.

Policies and measure. Let

$$\Pi = \prod_{\omega \in \Omega} \Delta(\mathcal{A}), \qquad \Pi^\circ = \mathrm{ri}(\Pi), \qquad d = |\Omega|(|\mathcal{A}| - 1).$$

We use Lebesgue measure in d independent policy coordinates, equivalently Lebesgue measure on the affine hull of Π. The interior is relative to that affine hull; Π has empty interior in the space of unrestricted probability matrices. Policies specify the conditional action law given the *full pre-action history*: $a_t \sim \beta(\cdot \mid \omega_t)$. They are not merely marginal approximations to a different history-dependent behaviour law. The POMDP, policies, traces, and sampling law remain fixed when comparing the operator at two value functions.

A specified starting posterior. For each behaviour policy μ, supply a distribution $w_\mu(z \mid \omega, a)$ on $g^{-1}(\omega)$ for every evaluated pair (ω, a). It is held fixed as the target policy varies. On the full evaluation domain define

$$J_{(z,a),(\omega,b)} = \mathbf{1}\{g(z) = \omega,\ a = b\}, \tag{11}$$

$$(W_\mu)_{(\omega,a),(z,b)} = \mathbf{1}\{a = b\}\, w_\mu(z \mid \omega, a). \tag{12}$$

Both matrices are nonnegative with row sums one, and $W_\mu J = I$. The conditional expectation $\mathbb{E}_{\mu, W_\mu}[\cdot \mid \omega_0 = \omega, a_0 = a]$ starts with $z_0 \sim w_\mu(\cdot \mid \omega, a)$ and the *fixed* initial action $a_0 = a$; subsequent transitions and actions follow $T_{\mathcal{Z}}$ and μ. The main text omits W_μ from the subscript of the expectation, leaving the starting posterior implicit. The same posterior is used when comparing target policies or trace parameters. For an unobserved initial pair, specifying this mathematical operator requires a posterior and the associated transition/reward law; it does not mean these quantities can be recovered from the behaviour data. The common-support results below avoid such unsupported initial pairs.

Retrace coefficients. Use the standard clip-at-one choice of Munos et al. (2016), with an irrelevant zero-probability convention made explicit:

$$c^{\lambda}_{\mu,\pi}(\omega,a)=\begin{cases}\lambda\min\left\{1,\dfrac{\pi(a\mid\omega)}{\mu(a\mid\omega)}\right\}, & \mu(a\mid\omega)>0,\\ 0, & \mu(a\mid\omega)=0,\end{cases}\qquad \lambda\in[0,1]. \tag{13}$$

The convention on a behaviour-impossible continuation action has no sample-path effect. In particular,

$$\mu(a\mid\omega)c^{\lambda}_{\mu,\pi}(\omega,a)=\lambda\min\{\mu(a\mid\omega),\pi(a\mid\omega)\}. \tag{14}$$

This pointwise-minimum representation of expected Retrace is also used in Section 4 of Munos et al. (2016). Define

$$\delta^{\pi}_{t}(q)=r_{t+1}+\gamma\sum_{b}\pi(b\mid\omega_{t+1})q(\omega_{t+1},b)-q(\omega_t,a_t), \tag{15}$$

$$\begin{aligned}(\mathcal{R}^{\lambda}_{\mu,\pi}q)(\omega,a)&=q(\omega,a)\\&\quad+\mathbb{E}_{\mu,W_\mu}\left[\sum_{t=0}^{\infty}\gamma^t\left(\prod_{u=1}^{t}c^{\lambda}_{\mu,\pi}(\omega_u,a_u)\right)\delta^{\pi}_{t}(q)\,\middle|\,\omega_0=\omega,a_0=a\right].\end{aligned} \tag{16}$$

The empty product is one. There is no initial-action ratio and no additional outer λ-mixture: λ is already in each trace coefficient. Equation 16 is the operator being analyzed; clipping ratios directly inside an n-step reward return is not an alternative definition of it.

For distinct $\lambda_1,\lambda_2\in[0,1]$, write

$$Q^{\lambda}_{\mu,\pi}=\mathrm{Fix}(\mathcal{R}^{\lambda}_{\mu,\pi}),\qquad D^{\lambda_1,\lambda_2}_{\mu,\pi}=\left\|D^{\lambda_1,\lambda_2}_{\mu,\pi}(\cdot,\cdot)\right\|, \tag{17}$$

where the pointwise discrepancy is defined in equation 6. Existence is proved next. The norm is any fixed norm on the retained value coordinates. A weighted norm must give positive weight to each such coordinate; this is the same nondegeneracy issue noted by Allen et al. (2024, Section 3.2 and Appendix G.2). The proofs apply to the squared Euclidean difference, whose zero set agrees with every such norm. Off policy, $D_{\mu,\pi}$ is not identified with the on-policy λ-discrepancy.

A.2 Projected contraction and a uniform fixed-point bound

Define latent state–action matrices

$$(P^{\beta})_{(z,a),(z',b)}=T_{\mathcal{Z}}(z'\mid z,a)\beta(b\mid g(z')), \tag{18}$$

$$C^{\lambda}_{\mu,\pi}=\mathrm{diag}_{(z,a)}\,c^{\lambda}_{\mu,\pi}(g(z),a),\qquad K^{\lambda}_{\mu,\pi}=P^{\mu}C^{\lambda}_{\mu,\pi}, \tag{19}$$

$$H^{\lambda}_{\mu,\pi}=(I-\gamma K^{\lambda}_{\mu,\pi})^{-1}. \tag{20}$$

Equivalently,

$$(K^{\lambda}_{\mu,\pi})_{(z,a),(z',b)}=\lambda T_{\mathcal{Z}}(z'\mid z,a)\min\{\mu(b\mid g(z')),\pi(b\mid g(z'))\}. \tag{21}$$

The next argument is a projected version of the positive-weight and row-sum argument in Munos et al. (2016, Theorem 1); the projection does not require a Bellman identity on the non-Markov representation.

We recall the theorem from the main text:

Theorem 1 (Contraction under partial observability)**.** *For fixed policies $\mu,\pi\in\Pi$ and a fixed starting-state sampling law, the operator $\mathcal{R}^{\lambda}_{\mu,\pi}$ in equation 5 is a γ-contraction in the supremum norm and has a unique fixed point. More generally, the same conclusion holds when the trace coefficient is replaced by any fixed nonnegative function c satisfying*

$$0\le\mu(a\mid\omega)c(\omega,a)\le\pi(a\mid\omega)\quad\textit{for every }(\omega,a).$$

Proof. More explicitly, we will prove

$$\mathcal{R}^{\lambda}_{\mu,\pi}q=b^{\lambda}_{\mu,\pi}+L^{\lambda}_{\mu,\pi}q,\qquad b^{\lambda}_{\mu,\pi}=W_\mu H^{\lambda}_{\mu,\pi}r,$$

$$L^{\lambda}_{\mu,\pi}=\gamma W_\mu H^{\lambda}_{\mu,\pi}(P^{\pi}-K^{\lambda}_{\mu,\pi})J, \tag{22}$$

$$Q^{\lambda}_{\mu,\pi}=(I-L^{\lambda}_{\mu,\pi})^{-1}b^{\lambda}_{\mu,\pi}, \tag{23}$$

$$\|Q^{\lambda}_{\mu,\pi}\|_\infty\le\frac{R_{\max}}{1-\gamma}. \tag{24}$$

Suppress the policy and trace indices. Domination gives $0 \le K \le P^\pi$ and $K\mathbf{1} \le \mathbf{1}$, so

$$H = \sum_{t\ge 0} \gamma^t K^t, \qquad \mathbf{1} \le H\mathbf{1} \le (1-\gamma)^{-1}\mathbf{1}.$$

For bounded q, the expected sum of absolute weighted residuals is at most $(R_{\max}+(1+\gamma)\|q\|_\infty)/(1-\gamma)$. Conditioning residuals on latent state–action pairs is therefore valid and yields

$$\mathcal{R}q = q + W_\mu H[r + (\gamma P^\pi - I)Jq].$$

Use $W_\mu J = I$ and $H - I = \gamma HK$ to obtain equation 22. Since $P^\pi - K \ge 0$, $L \ge 0$. Moreover,

$$L\mathbf{1} = \gamma W_\mu H(P^\pi - K)\mathbf{1} = \mathbf{1} - (1-\gamma)W_\mu H\mathbf{1} \le \gamma\mathbf{1}. \tag{25}$$

Thus $\|L\|_\infty \le \gamma$, proving contraction. The Banach fixed-point theorem gives a unique fixed point, and the convergent matrix geometric series gives $(I-L)^{-1} = \sum_{j\ge 0} L^j \ge 0$. Finally, $|b| \le R_{\max} W_\mu H\mathbf{1}$ and $(I-L)\mathbf{1} = (1-\gamma)W_\mu H\mathbf{1}$. Consequently,

$$|Q| \le (I-L)^{-1}|b| \le R_{\max}(I-L)^{-1}W_\mu H\mathbf{1} = \frac{R_{\max}}{1-\gamma}\mathbf{1},$$

which proves equation 24. □

A.3 Common support faces and matched on-policy posteriors

Reachability and the evaluated coordinates. Fix one initial law ζ, independent of policies and support choices. For every observation prescribe a nonempty set $S(\omega) \subseteq \mathcal{A}$. Let $\mathcal{Z}_S$ contain the latent states reachable from ζ using only actions in $S(g(z))$, and put $\Omega_S = g(\mathcal{Z}_S)$. This is the same reachable set under every policy positive on all allowed actions: every finite admissible path has positive probability. It is closed under allowed transitions. Define

$$\mathcal{F}_S = \prod_{\omega\in\Omega_S} \Delta(S(\omega)), \qquad \mathcal{F}_S^\circ = \mathrm{ri}(\mathcal{F}_S), \tag{26}$$

$$d_S = \sum_{\omega\in\Omega_S} (|S(\omega)| - 1),$$

$$\mathcal{I}_S = \{(\omega,a) : \omega \in \Omega_S,\ a \in S(\omega)\}, \qquad \mathcal{E}_S = \{(z,a) : z \in \mathcal{Z}_S,\ a \in S(g(z))\}. \tag{27}$$

A policy is extended by zero on excluded actions at retained representations. Unreachable representations are discarded, so $\mathcal{F}_S$ is a face of the policy polytope on Ω_S. Keeping unused policy rows would only append irrelevant product factors. Let Leb_S be Lebesgue measure in d_S independent coordinates. When $d_S = 0$, the face is a singleton; for product-measure formulas one may assign that point mass one. Positive-dimensional almost-everywhere conclusions will be stated explicitly as such.

Restricted operators. Use value vectors in $\mathbb{R}^{\mathcal{I}_S}$ and latent vectors in $\mathbb{R}^{\mathcal{E}_S}$. Define $J_S, P_S^\beta, K_{\mu,\pi,S}^\lambda$ by restriction of the earlier formulas, and choose $W_{\mu,S}$ supported on $\mathcal{Z}_S \cap g^{-1}(\omega)$ for every $(\omega,a) \in \mathcal{I}_S$. There is no row renormalization: P_S^β is stochastic, and $W_{\mu,S}J_S = I$. This is a closed evaluation problem. Write

$$\begin{aligned} Q_{\mu,\pi,S}^\lambda &= \mathrm{Fix}(\mathcal{R}_{\mu,\pi,S}^\lambda), \\ D_{\mu,\pi,S}^{\lambda_1,\lambda_2}(\omega,a) &= \left| Q_{\mu,\pi,S}^{\lambda_1}(\omega,a) - Q_{\mu,\pi,S}^{\lambda_2}(\omega,a) \right|, \\ D_{\mu,\pi,S}^{\lambda_1,\lambda_2} &= \left\| \left(D_{\mu,\pi,S}^{\lambda_1,\lambda_2}(\omega,a)\right)_{(\omega,a)\in\mathcal{I}_S} \right\|_S, \end{aligned} \tag{28}$$

where $\|\cdot\|_S$ gives positive weight to all retained coordinates. Theorem 1 applies. If $\mu \in \mathcal{F}_S^\circ$, it covers every $\pi \in \mathcal{F}_S$, including targets on the relative boundary.

Discounted-occupancy sampling. To connect to the on-policy discrepancy we use the posterior of Allen et al. (2024, Section 3 and Appendix C), not an arbitrary replay-window posterior. For $\beta \in \mathcal{F}_S^\circ$,

let

$$B_S^\beta(z, z') = \sum_{a \in S(g(z))} \beta(a \mid g(z)) T_{\mathcal{Z}}(z' \mid z, a), \tag{29}$$

$$d_{\beta,S} = (1-\gamma)\zeta_S (I - \gamma B_S^\beta)^{-1}, \tag{30}$$

$$w_{\beta,S}(z \mid \omega, a) = \frac{\mathbf{1}\{g(z) = \omega\}\, d_{\beta,S}(z)}{\sum_{y: g(y)=\omega} d_{\beta,S}(y)}. \tag{31}$$

Here ζ_S is the restriction of ζ, still of total mass one. All denominators are positive on the retained set. The initial action cancels from the posterior because it is chosen conditional only on ω. For behaviour μ, construct $W_{\mu,S}$ using this posterior and hold it fixed while varying π. For the full-action support these formulas are denoted d_β, W_β without S.

$$d_\beta = (1-\gamma)\zeta (I - \gamma B^\beta)^{-1}, \tag{32}$$

$$w_\beta(z \mid \omega, a) = \frac{\mathbf{1}\{g(z) = \omega\} d_\beta(z)}{d_\beta^\Omega(\omega)}. \tag{33}$$

The full model is henceforth restricted to states reachable using all actions, so its posterior is defined for every $\beta \in \Pi^\circ$.

Lemma 4 (On-policy identification and analyticity on a face)**.** *For $\beta \in \mathcal{F}_S^\circ$, with the matched posterior above, $\mathcal{R}^\lambda_{\beta,\beta,S}$ is the on-policy TD(λ) return operator. Its fixed point is the corresponding on-policy value $Q^\lambda_{\beta,S}$, and*

$$\begin{aligned} D^{\lambda_1,\lambda_2}_{\beta,\beta,S}(\omega, a) = \Lambda^{\lambda_1,\lambda_2}_{\beta,S}(\omega, a) &:= \left| Q^{\lambda_1}_{\beta,S}(\omega, a) - Q^{\lambda_2}_{\beta,S}(\omega, a) \right|, \\ D^{\lambda_1,\lambda_2}_{\beta,\beta,S} = \Lambda^{\lambda_1,\lambda_2}_{\beta,S} &:= \left\| \left(\Lambda^{\lambda_1,\lambda_2}_{\beta,S}(\omega, a)\right)_{(\omega,a)\in\mathcal{I}_S} \right\|_S. \end{aligned} \tag{34}$$

Both $W_{\beta,S}$ and $Q^\lambda_{\beta,S}$ are real analytic in the independent coordinates of $\mathcal{F}_S^\circ$.

Proof. The entries of B_S^β are affine, its discounted resolvent is nonsingular, and all posterior denominators are positive. Matrix inversion and division therefore make $W_{\beta,S}$ rational and real analytic. This is the finite-matrix analytic strategy of Allen et al. (2024, Appendix D). On the diagonal all retained coefficients equal λ. Writing $H^\lambda_{\beta,S} = H^\lambda_{\beta,\beta,S}$, and similarly for b and L, gives

$$\begin{aligned} H^\lambda_{\beta,S} &= (I - \gamma\lambda P_S^\beta)^{-1}, \qquad b^\lambda_{\beta,S} = W_{\beta,S} H^\lambda_{\beta,S} r_S, \\ L^\lambda_{\beta,S} &= \gamma(1-\lambda) W_{\beta,S} H^\lambda_{\beta,S} P_S^\beta J_S. \end{aligned} \tag{35}$$

For $\lambda < 1$, expanding the resolvent gives

$$\begin{aligned} &(\mathcal{R}^\lambda_{\beta,\beta,S} q)(\omega, a) \\ &\quad = (1-\lambda) \sum_{n \geq 1} \lambda^{n-1} \mathbb{E}_{\beta, W_{\beta,S}} \left[\sum_{t=0}^{n-1} \gamma^t r_{t+1} + \gamma^n q(\omega_n, a_n) \,\middle|\, \omega_0 = \omega, a_0 = a \right]. \end{aligned}$$

This is the TD(λ) return mixture (Sutton, 1988), with the POMDP posterior used by Allen et al. (2024, Appendix C). For $\lambda = 1$, $L^\lambda_{\beta,S} = 0$ and the operator is the expected Monte Carlo return. Uniqueness identifies the fixed points. Finally, equation 23 with equation 35 is rational with nonsingular inverses throughout the relative interior. □

A.4 Genericity inside a fixed support class

The analytic input is the following classical fact: a nontrivial real-analytic scalar function on a connected open subset of $\mathbb{R}^n$, $n \geq 1$, has a Lebesgue-null zero set (Mityagin, 2020). Allen et al. (2024, Appendix D) use this fact for the on-policy discrepancy. Hard clipping requires a cellwise argument for a fixed off-policy behaviour.

Fix $\mu \in \mathcal{F}_S^\circ$ and its posterior (arbitrary but fixed in the next two results). Partition $\mathcal{F}_S^\circ$ using

$$\mathcal{H}_{\mu,S} = \bigcup_{\omega \in \Omega_S:\ |S(\omega)| \geq 2} \ \bigcup_{a \in S(\omega)} \{\pi : \pi(a \mid \omega) = \mu(a \mid \omega)\}.$$

Constant coordinates are omitted: excluded probabilities are zero and singleton allowed actions have probability one. All included equalities are proper relative hyperplanes. The nonempty connected components of $\mathcal{F}_S^\circ \setminus \mathcal{H}_{\mu,S}$ are finitely many open convex *clipping cells* in face coordinates.

Lemma 5 (Continuity and cellwise analyticity)**.** *For fixed $\mu, W_{\mu,S}$, the map $\pi \mapsto Q^{\lambda}_{\mu,\pi,S}$ is continuous on $\mathcal{F}_S$ and real analytic on every clipping cell. The scalar function of π defined by*

$$f^{\lambda_1,\lambda_2}_{\mu,\pi,S} = \sum_{(\omega,a)\in\mathcal{I}_S} \big(Q^{\lambda_1}_{\mu,\pi,S}(\omega,a) - Q^{\lambda_2}_{\mu,\pi,S}(\omega,a)\big)^2$$

has these same regularity properties and is zero exactly where $D^{\lambda_1,\lambda_2}_{\mu,\pi,S}$ is zero.

Proof. The kernel equation 21 is continuous on the entire face and affine on each clipping cell. The inverses in equation 20 and equation 23 are nonsingular by contraction. Continuity of inversion and its adjugate/determinant formula give continuity on the face and rational, hence analytic, coordinates on each cell, as in the matrix-analytic method of Allen et al. (2024, Appendix D). Squaring and summing gives the last assertion. □

Lemma 6 (Cellwise genericity)**.** *Assume $d_S \geq 1$. If $D^{\lambda_1,\lambda_2}_{\mu,\pi_0,S} > 0$ for some π_0 in a clipping cell $\mathcal{C}$, then*

$$\mathrm{Leb}_S\{\pi \in \mathcal{C} : D^{\lambda_1,\lambda_2}_{\mu,\pi,S} = 0\} = 0.$$

Proof. Lemma 5 makes $\pi \mapsto f^{\lambda_1,\lambda_2}_{\mu,\pi,S}$ a nontrivial analytic function on the connected open set $\mathcal{C}$. Apply the analytic zero-set theorem (Mityagin, 2020). □

Theorem 7 (Support-relative genericity from a diagonal witness)**.** *Let $d_S \geq 1$, $\mu \in \mathcal{F}_S^\circ$, and fix its supported posterior. If $D^{\lambda_1,\lambda_2}_{\mu,\mu,S} > 0$, then*

$$\mathrm{Leb}_S\{\pi \in \mathcal{F}_S : D^{\lambda_1,\lambda_2}_{\mu,\pi,S} = 0\} = 0.$$

The target need not belong to the relative interior of the face.

Proof. Choose $\bar\pi$ in any clipping cell and set $\pi_t = (1-t)\mu + t\bar\pi$ for $0 < t \leq 1$. For every nonconstant coordinate,

$$\pi_t(a \mid \omega) - \mu(a \mid \omega) = t\big(\bar\pi(a \mid \omega) - \mu(a \mid \omega)\big).$$

Thus π_t stays in that cell and approaches μ. Continuity and the diagonal witness supply a nonzero witness in every cell. Lemma 6 makes each cell's zero set null. Take the finite union and add $\mathcal{H}_{\mu,S}$ and the relative boundary of $\mathcal{F}_S$, both null under Leb_S. No analyticity on the boundary is asserted or needed. □

Corollary 8 (All target policies in the full policy space)**.** *For full-action support, fix $\mu \in \Pi^\circ$ and its supported posterior. Then*

$$D^{\lambda_1,\lambda_2}_{\mu,\mu} > 0 \quad\Longrightarrow\quad \mathrm{Leb}_d\{\pi \in \Pi : D^{\lambda_1,\lambda_2}_{\mu,\pi} = 0\} = 0.$$

Proof. Take $S(\omega) = \mathcal{A}$ in Theorem 7. □

Corollary 9 (Generic on-policy and off-policy detection on one face)**.** *Use the discounted-occupancy posterior and let $d_S \geq 1$. If*

$$\exists \beta_0 \in \mathcal{F}_S^\circ : \Lambda^{\lambda_1,\lambda_2}_{\beta_0,S} > 0,$$

then the on-policy zero set

$$E_S = \{\beta \in \mathcal{F}_S^\circ : \Lambda^{\lambda_1,\lambda_2}_{\beta,S} = 0\}$$

is Leb_S-null. For every $\mu \in \mathcal{F}_S^\circ \setminus E_S$,

$$\mathrm{Leb}_S\{\pi \in \mathcal{F}_S : D^{\lambda_1,\lambda_2}_{\mu,\pi,S} = 0\} = 0.$$

The joint zero set in $\mathcal{F}_S^\circ \times \mathcal{F}_S$ is null under $\mathrm{Leb}_S \otimes \mathrm{Leb}_S$.

Proof. The squared Euclidean on-policy difference is analytic on the connected relative interior by Lemma 4, and is nontrivial by the witness. The zero-set theorem proves that E_S is null, reproducing the argument of Allen et al. (2024, Theorem 1 and Appendix D) on this restricted process. Outside E_S, the diagonal identification and Theorem 7 give the target-policy conclusion. The joint zero set is measurable because the fixed points are jointly continuous for $\mu \in \mathcal{F}_S^\circ$, $\pi \in \mathcal{F}_S$. Tonelli–Fubini applied to its indicator proves the product-measure claim. □

A.5 A WITNESS SURVIVES ENLARGEMENT OF ITS ACTION SUPPORT

The preceding theorem does not require a new independent witness for every larger support. We now prove the upward implication using compatible posteriors. Throughout this subsection, the transition and reward laws, observation map g, initial law ζ, discount, and trace parameters remain the same across support restrictions. Only the allowed actions are changed.

Lemma 10 (Continuity on retained coordinates under support enlargement)**.** *Let $S(\omega) \subseteq U(\omega)$ for every observation, and let $\beta_0 \in \mathcal{F}_S^\circ$, allowing $d_S = 0$. Extend β_0 to $\bar{\beta}_0 \in \mathcal{F}_U$ by assigning zero probability to $U(\omega) \setminus S(\omega)$ on Ω_S and choosing any policy on $\Omega_U \setminus \Omega_S$. For any $\eta \in \mathcal{F}_U^\circ$ set $\beta_\varepsilon = (1-\varepsilon)\bar{\beta}_0 + \varepsilon\eta$, $0 < \varepsilon < 1$. With the matched occupancy posteriors,*

$$Q^\lambda_{\beta_\varepsilon,\beta_\varepsilon,U}\big|_{\mathcal{I}_S} \longrightarrow Q^\lambda_{\beta_0,\beta_0,S} \quad \text{as } \varepsilon \downarrow 0, \qquad \lambda \in [0,1]. \tag{36}$$

The limit is independent of the extension on newly reachable representations. No limiting posterior is required at those new representations.

Proof. Work on the fixed enlarged latent space $\mathcal{Z}_U$. The matrices $B_U^{\beta_\varepsilon}$ converge to $B_U^{\bar{\beta}_0}$, and their discounted resolvents are nonsingular. Thus

$$d_{\beta_\varepsilon,U} \longrightarrow d_0 := (1-\gamma)\zeta_U(I - \gamma B_U^{\bar{\beta}_0})^{-1}.$$

The set $\mathcal{Z}_S$ is closed under $\bar{\beta}_0$ and contains the support of ζ. Hence d_0 is the zero extension of $d_{\beta_0,S}$. At every old observation $\omega \in \Omega_S$, the limiting denominator in the posterior is positive. Therefore the old rows of $W_{\beta_\varepsilon,U}$ converge to the old posterior rows, zero-extended to $\mathcal{E}_U$.

Partition value coordinates into $O = \mathcal{I}_S$ and $N = \mathcal{I}_U \setminus \mathcal{I}_S$, and suppress the fixed λ. Let $q_\varepsilon = b_\varepsilon + L_\varepsilon q_\varepsilon$ be the enlarged diagonal fixed-point equation and $q_S = b_S + L_S q_S$ the restricted one. The latent state–action set $\mathcal{E}_S$ is closed under $P_U^{\bar{\beta}_0}$: an old initial action is allowed in S, and the next action again lies in S. The diagonal formulas equation 35, continuity of $(I - \gamma\lambda P_U^{\beta_\varepsilon})^{-1}$, and convergence of the old posterior rows now give

$$b_{\varepsilon,O} \to b_S, \qquad L_{\varepsilon,OO} \to L_S, \qquad L_{\varepsilon,ON} \to 0. \tag{37}$$

In particular, the old rows of the limiting latent resolvent have zero entries in new latent columns; multiplication by J_U cannot create an old-to-new value transition. This proves the last limit without assuming convergence of any new posterior row.

By Theorem 1, $\|L_{\varepsilon,OO}\|_\infty \le \gamma$ and $\|q_\varepsilon\|_\infty \le R_{\max}/(1-\gamma)$, uniformly in ε. Subtract the old-coordinate equations to obtain

$$\begin{aligned}\|q_{\varepsilon,O} - q_S\|_\infty \le \frac{1}{1-\gamma}\Big(&\|b_{\varepsilon,O} - b_S\|_\infty + \|L_{\varepsilon,OO} - L_S\|_\infty\|q_S\|_\infty \\ &+ \|L_{\varepsilon,ON}\|_\infty\|q_{\varepsilon,N}\|_\infty\Big) \longrightarrow 0.\end{aligned} \tag{38}$$

The empty new block is interpreted as zero. This establishes the claim. □

Theorem 11 (Support enlargement and generic detection)**.** *Under the matched sampling convention, suppose $\beta_0 \in \mathcal{F}_S^\circ$ satisfies $\Lambda^{\lambda_1,\lambda_2}_{\beta_0,S} > 0$. Then for every $U \supseteq S$ there is a policy in $\mathcal{F}_U^\circ$ with nonzero on-policy discrepancy. If $d_U \ge 1$, there is a Leb_U-null set $E_U \subset \mathcal{F}_U^\circ$ such that*

$$\Lambda^{\lambda_1,\lambda_2}_{\beta,U} > 0 \qquad \text{for every } \beta \in \mathcal{F}_U^\circ \setminus E_U, \tag{39}$$

$$\mathrm{Leb}_U\{\pi \in \mathcal{F}_U : D^{\lambda_1,\lambda_2}_{\mu,\pi,U} = 0\} = 0 \qquad \text{for every } \mu \in \mathcal{F}_U^\circ \setminus E_U. \tag{40}$$

The joint off-policy zero set in $\mathcal{F}_U^\circ \times \mathcal{F}_U$ also has product measure zero.

Proof. The nonzero witness differs on at least one retained coordinate $(\omega_0, a_0) \in \mathcal{I}_S$. Apply Lemma 10 to λ_1 and λ_2. Their difference at this coordinate remains nonzero for all sufficiently small $\varepsilon > 0$. Thus $\beta_\varepsilon \in \mathcal{F}_U^\circ$ supplies an interior witness in the enlarged class. Apply Corollary 9 with support U. If $d_U = 0$, existence simply concerns the only policy in that class. □

Remark 12 (Upward inheritance, not a monotonicity or minimality claim). Support patterns admitting a supported on-policy witness are upward closed. This is an existence statement, not monotonicity of discrepancy magnitude, and it does not identify a smallest informative action set. No corresponding downward implication follows: deleting a needed action can remove the witness. An arbitrary off-policy witness with a fixed behaviour is not the premise of Theorem 11. The lifting is on policy; the clipping-cell argument is applied only after a diagonal witness has been obtained in the larger class.

An arbitrary policy witness, including boundary policies. For $\beta \in \Pi$ define $S_\beta(\omega) = \operatorname{supp} \beta(\cdot \mid \omega)$. Construct its reachable restricted process as above. Its restriction belongs to $\mathcal{F}_{S_\beta}^\circ$, even when the original policy is on the boundary of Π. Define the *supported on-policy discrepancy* by

$$\Lambda_\beta^{\lambda_1,\lambda_2} := \Lambda_{\beta,S_\beta}^{\lambda_1,\lambda_2}. \tag{41}$$

On the right, β is restricted to Ω_{S_β}. For an interior policy this is the usual full-action discrepancy. On the boundary, it never uses values at unreachable observations or excluded initial actions. A posterior at a zero-occupancy observation is not defined by an arbitrary convention.

Theorem 3 (Generic off-policy detection)**.** *Fix distinct $\lambda_1, \lambda_2 \in [0, 1]$ and use the sampling convention of Appendix A.3 with a common initial distribution. Suppose there exists $\beta_0 \in \Pi$ whose supported on-policy discrepancy satisfies $\Lambda_{\beta_0}^{\lambda_1,\lambda_2} > 0$. Then there is a Lebesgue-null set $E \subset \Pi$, containing the relative boundary of Π, such that, for every $\mu \in \Pi \setminus E$,*

$$\mathrm{Leb}_d\{\pi \in \Pi : D_{\mu,\pi}^{\lambda_1,\lambda_2} = 0\} = 0.$$

Proof. Use $S = S_{\beta_0}$ and $U(\omega) = \mathcal{A}$ in Theorem 11. This proves the interior on-policy claim and the full target-policy off-policy claim. The remaining on-policy zero set lies on the null boundary of Π; every subset of that boundary is Lebesgue measurable and null. □

This recovers the one-witness/almost-everywhere *on-policy* conclusion of Allen et al. (2024, Theorem 1) in the present finite discounted setting, with the supported-coordinate convention explicit. The off-policy conclusion is established separately. It cannot be deduced on the diagonal merely from a product-almost-everywhere result, since the diagonal is itself a product-null subset when $d > 0$.

A.6 Interpretation, counterexample, and scope

Corollary 2 (Vanishing for Markov-sufficient observations)**.** *Suppose ω is a Markov-sufficient observation of the controlled process, so that its reward and observation-transition laws depend only on (ω, a). Then $Q_{\mu,\pi}^\lambda = Q_\pi$ for every behaviour/target pair and every $\lambda \in [0, 1]$. Hence $D_{\mu,\pi}^{\lambda_1,\lambda_2}(\omega, a) = 0$ at every evaluated pair, and $D_{\mu,\pi}^{\lambda_1,\lambda_2} = 0$ for all λ_1, λ_2.*

Proof. This is the usual Retrace fixed-point argument (Munos et al., 2016, Theorem 1), also underlying the zero-discrepancy property of Markov representations in Allen et al. (2024). The induced MDP's Bellman equation implies $r + (\gamma P^\pi - I)JQ_\pi = 0$ on latent pairs. The residual representation gives $\mathcal{R}_{\mu,\pi}^\lambda Q_\pi = Q_\pi$, and contraction gives uniqueness. □

Why an arbitrary fixed-behaviour witness is insufficient. The following explicit example is included as part of our analysis, not as a result attributed to the cited papers. There are three hidden states, one observation, two actions, and $\gamma = 1/2$. Specify

	$T_A(s)$	$r_A(s)$	$T_B(s)$	$r_B(s)$
s_1	s_2	$-1/2$	s_1	0
s_2	s_3	3	s_1	0
s_3	s_1	-4	s_1	0

Take $\mu(A) = \mu(B) = 1/2$ and $\zeta = (4/7, 2/7, 1/7)$, the stationary hidden-state distribution under μ. Its discounted posterior is therefore the same vector. Let $\pi_p = (p, 1-p)$, $0 < p < 1$, and use $\lambda_1 = 0, \lambda_2 = 1$. Since $\zeta r_A = \zeta r_B = 0$, $Q^0_{\mu,\pi_p} = (0, 0)$. Substitution into equation 23 yields

$$Q^1_{\mu,\pi_p} = \begin{cases} \dfrac{2p(1-2p)}{p^2+2p+4} \begin{pmatrix} p \\ p-1 \end{pmatrix}, & 0 < p < 1/2, \\ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, & 1/2 \le p < 1. \end{cases} \tag{42}$$

The substitution uses trace-weighted action probabilities $(p, 1/2)$ on the left interval and $(1/2, 1-p)$ on the right interval. For example, the value at $p = 1/4$ is $(1/73, -3/73)$, yet the discrepancy is zero on an interval of positive measure. This does not contradict the diagonal-witness theorem: $D_{\mu,\mu} = 0$.

Support and information limits. A null set in the full simplex can contain an entire proper face. The support-relative results are informative on a larger prescribed face when that face contains a witnessed support, but give no downward implication or guarantee for a specified deterministic target. A deterministic support class has no policy variation, although its witness can be lifted to larger classes. If target and behaviour supports are disjoint at every representation, equation 21 gives $K^\lambda_{\mu,\pi} = 0$, so

$$\mathcal{R}^\lambda_{\mu,\pi} q = W_\mu r + \gamma W_\mu P^\pi J q$$

is independent of λ. The discrepancy is then zero regardless of partial observability. This is trace truncation, not evidence of sufficiency. The common-support assumptions exclude this situation.

What is and is not guaranteed. The witness premise is essential and reward-dependent; not every non-Markov representation must produce a nonzero discrepancy. The results concern exact fixed points for a *fixed* finite POMDP and tabular observation-based policies. They give no lower bound on discrepancy magnitude, no converse to the sufficient-Markov condition, and no convergence theorem for learning policies or observation representations. Policy-coordinate Lebesgue measure does not automatically transfer to arbitrary neural-network parameterizations or training trajectories. Full behaviour support alone does not remove exceptional behaviours; a fixed behaviour needs its diagonal witness. Contraction of an expected operator must also be distinguished from sample-based convergence with visitation assumptions (Munos et al., 2016, Theorem 3).

Additional future-dependent regression weights are not part of equation 16. For a detached nonnegative weight A and a fixed target Y_q, exact scalar weighted least squares yields

$$\frac{\mathbb{E}[AY_q \mid \omega_0 = \omega, a_0 = a]}{\mathbb{E}[A \mid \omega_0 = \omega, a_0 = a]},$$

when the denominator is positive, rather than $\mathbb{E}[Y_q \mid \omega_0 = \omega, a_0 = a]$. This identity follows by differentiating the conditional quadratic loss; such a weighted update requires its own operator analysis. Bounded trace coefficients alone also do not bound derivatives of a neural function approximator or prove a gradient-variance guarantee.

A.7 Finite-action entropy-regularized values

The main derivation uses ordinary rewards. For completeness, the expected soft Bellman bootstrap of Haarnoja et al. (2018, Section 4) can be included without changing the trace argument. This extension concerns a finite action space and a fixed finite temperature $\alpha \ge 0$. Define $h_\pi(\omega) = -\sum_a \pi(a \mid \omega) \log \pi(a \mid \omega)$, with $0 \log 0 = 0$, and replace the residual by

$$\delta^{\pi,\alpha}_t(q) = r_{t+1} + \gamma \sum_b \pi(b \mid \omega_{t+1}) q(\omega_{t+1}, b) + \gamma \alpha h_\pi(\omega_{t+1}) - q(\omega_t, a_t). \tag{43}$$

The initial action has no entropy bonus; this is the action-value convention corresponding to the soft next-state bootstrap.

Theorem 13 (Extension to the expected-entropy residual)**.** *All contraction, face-genericity, support-enlargement, and one-supported- witness conclusions above hold for equation 43, with both discrepancies and the witness interpreted at the* same *temperature α. The fixed-point bound becomes*

$$\|Q^{\lambda,\alpha}_{\mu,\pi}\|_\infty \le \frac{R_{\max} + \gamma\alpha \log |\mathcal{A}|}{1-\gamma}.$$

The Markov-sufficient conclusion holds with the corresponding soft policy value in place of Q_π.

Proof. Conditional expectation replaces the reward vector by

$$r_{\pi,\alpha}(z,a) = r(z,a) + \gamma\alpha \sum_{z'} T_{\mathcal{Z}}(z' \mid z,a) h_\pi(g(z')).$$

For fixed π this is independent of q, so L is unchanged and $b = W_\mu H r_{\pi,\alpha}$. The finite-action bounded-reward reduction is also used by Haarnoja et al. (2018, Lemma 1). More explicitly, concavity of $\log$ gives

$$h_\pi(\omega) = \sum_{a:\pi(a|\omega)>0} \pi(a \mid \omega) \log \frac{1}{\pi(a \mid \omega)} \le \log |\operatorname{supp} \pi(\cdot \mid \omega)| \le \log |\mathcal{A}|.$$

Together with $h_\pi \ge 0$, this proves the uniform bound. On the relative interior of a support face, h_π is real analytic; its extension to the closed face is continuous because $t \log t \to 0$ as $t \downarrow 0$. Fixed points are therefore analytic on clipping cells and on the on-policy relative interior, though no longer necessarily rational. The same zero-set arguments apply. For support enlargement, $r_{\beta_\varepsilon,\alpha}$ is uniformly bounded and converges on old latent rows to the restricted reward; closure of the old process excludes new-representation contributions in the limit. Thus the block proof of Lemma 10 continues to hold. Finally the induced soft Bellman equation proves vanishing under a sufficient representation as before. □

This is not an implication from an unregularized witness to a regularized one. It also does not establish the same policy-measure result for continuous-action policies, differential entropy, or continuous observation spaces; those require different spaces, measures, and regularity arguments.

B Parity Check Environment

We give the formal model and calculations supporting Section 3.4. We use the unregularized residual ($\alpha = 0$) and the supremum norm for exact discrepancy magnitudes; the zero-set conclusion holds for any norm.

Environment and notation. At the start of each episode, two independent signs $U, V \in \{-1, +1\}$ are sampled uniformly, selecting one of four equally likely hidden paths. The agent observes u_U at the first step, v_V at the second, and a white junction observation j at the third, where $u_+ = \text{red}$, $u_- = \text{blue}$, $v_+ = \text{pink}$, and $v_- = \text{cyan}$. Both actions advance along the selected path and give zero reward at the first two steps. At the junction, UP gives reward UV and DOWN gives reward $-UV$, after which the episode terminates with zero continuation value. We use this uniform initial law with the discounted-occupancy sampling convention of Appendix A.3. There are five nonterminal observations and two actions, so Π has five independent policy coordinates. We formulate the claim from the main text as follows.

Theorem 14 (Every fully supported behaviour policy detects Parity Check generically)**.** *For the parity check environment, fix $0 < \gamma < 1$ and distinct $\lambda_1, \lambda_2 \in [0,1]$. Every memoryless on-policy pair has $Q^\lambda_{\beta,\beta} = 0$ for every $\lambda \in [0,1]$ (on its supported evaluation domain for boundary policies). Nevertheless, for every $\mu \in \Pi^\circ$,*

$$\mathrm{Leb}_5\big\{\pi \in \Pi : D^{\lambda_1,\lambda_2}_{\mu,\pi} = 0\big\} = 0. \tag{44}$$

Proof. Define

$$\kappa_\pm = \sum_{a \in \{\mathrm{UP},\mathrm{DOWN}\}} \min\{\mu(a \mid v_\pm), \pi(a \mid v_\pm)\},$$
$$h = \min\{\mu(\mathrm{UP} \mid j), \pi(\mathrm{UP} \mid j)\} - \min\{\mu(\mathrm{DOWN} \mid j), \pi(\mathrm{DOWN} \mid j)\}. \tag{45}$$

We will prove that, for either initial action a,

$$Q^\lambda_{\mu,\pi}(u_U, a) = \frac{\gamma^2\lambda^2}{2} U(\kappa_+ - \kappa_-)h, \qquad Q^\lambda_{\mu,\pi}(v_\pm, a) = Q^\lambda_{\mu,\pi}(j, a) = 0, \tag{46}$$

and hence

$$D^{\lambda_1,\lambda_2}_{\mu,\pi} = \frac{\gamma^2}{2}|\lambda_1^2 - \lambda_2^2||\kappa_+ - \kappa_-|\,|h|. \tag{47}$$

Exact fixed points. The first two actions do not affect transitions, so every hidden path retains its initial probability under any memoryless behaviour policy. All paths reach each stage at the same time. Thus discounted-occupancy sampling preserves their relative weights, giving $\mathbb{E}[U \mid v_V] = 0$, $\mathbb{E}[V \mid u_U] = 0$, and $\mathbb{E}[UV \mid j] = 0$. The behaviour-weighted trace coefficients satisfy

$$\mu(a \mid x)c^\lambda(x,a) = \lambda \min\{\mu(a \mid x), \pi(a \mid x)\},$$

as in the expected trace kernel of Munos et al. (2016, Section 3.3). Consequently,

$$\mathbb{E}_\mu[c^\lambda \mid v_\pm] = \lambda\kappa_\pm, \qquad \mathbb{E}_\mu[c^\lambda r \mid U, V, j] = \lambda UVh.$$

Write $q^\lambda = Q^\lambda_{\mu,\pi}$. At j, either fixed action has expected immediate reward zero under the uniform parity posterior, so $q^\lambda(j,a) = 0$. Starting at v_V, the fixed initial action has zero reward, and the remaining fixed-point equation is

$$q^\lambda(v_V, a) = \gamma\mathbb{E}_\mu[c_1^\lambda r_2 \mid v_V, a] = \gamma\lambda Vh\,\mathbb{E}[U \mid v_V] = 0.$$

The bootstrap values in this equation are zero by the junction calculation. At the first stage, all later fixed-point values are now zero. Thus

$$\begin{aligned} q^\lambda(u_U, a) &= \gamma^2\mathbb{E}_\mu[c_1^\lambda c_2^\lambda r_3 \mid u_U, a] \\ &= \frac{\gamma^2}{2}\left[(\lambda\kappa_+)(\lambda Uh) + (\lambda\kappa_-)(-\lambda Uh)\right] \\ &= \frac{\gamma^2\lambda^2}{2}U(\kappa_+ - \kappa_-)h. \end{aligned}$$

This proves equation 46; taking the supremum norm of the difference between two horizons gives equation 47. For an on-policy pair, $\kappa_+ = \kappa_- = 1$, so every value is zero. For a boundary on-policy pair, use $c^\lambda = \lambda$ on sampled actions: the same backward calculation proves the assertion on supported coordinates. This recovers the on-policy cancellation of Allen et al. (2024).

The exceptional target policies. Fix $\mu \in \Pi^\circ$ and write

$$m_\pm = \mu(\mathrm{UP} \mid v_\pm), \quad p_\pm = \pi(\mathrm{UP} \mid v_\pm), \quad m_j = \mu(\mathrm{UP} \mid j), \quad p_j = \pi(\mathrm{UP} \mid j).$$

For binary actions, elementary identities give

$$\kappa_\pm = 1 - |m_\pm - p_\pm|, \qquad h = m_j + p_j - 1.$$

The prefactor in equation 47 is strictly positive. It follows that the zero set is exactly the intersection of Π with the union of three proper affine hyperplanes:

$$\{p_j = 1 - m_j\} \cup \{p_+ - p_- = m_+ - m_-\} \cup \{p_+ + p_- = m_+ + m_-\}. \tag{48}$$

The first-stage target coordinates are unrestricted. Each of these hyperplanes has zero five-dimensional Lebesgue measure, giving the desired result. This exceptional set is independent of λ_1, λ_2, so the conclusion holds simultaneously for all distinct trace parameters in $[0, 1]$. □

B.1 The explicit detection witness

The policies used in Section 3.4 are

x	$\mu(\mathrm{UP} \mid x)$	$\pi(\mathrm{UP} \mid x)$
red, blue	$1/2$	$1/2$
pink	$1/2$	$1/2$
cyan	$1/2$	$3/4$
white	$3/4$	$3/4$

(49)

with complementary probabilities for DOWN. The coefficients $c^\lambda = \lambda\min(1, \pi/\mu)$ equal λ at every observation–action pair except cyan–DOWN, where the coefficient is $\lambda/2$. The cyan–UP ratio $3/2$ is clipped to one, giving

$$\mathbb{E}_\mu[c^\lambda \mid \mathrm{pink}] = \lambda, \qquad \mathbb{E}_\mu[c^\lambda \mid \mathrm{cyan}] = \frac{3\lambda}{4}.$$

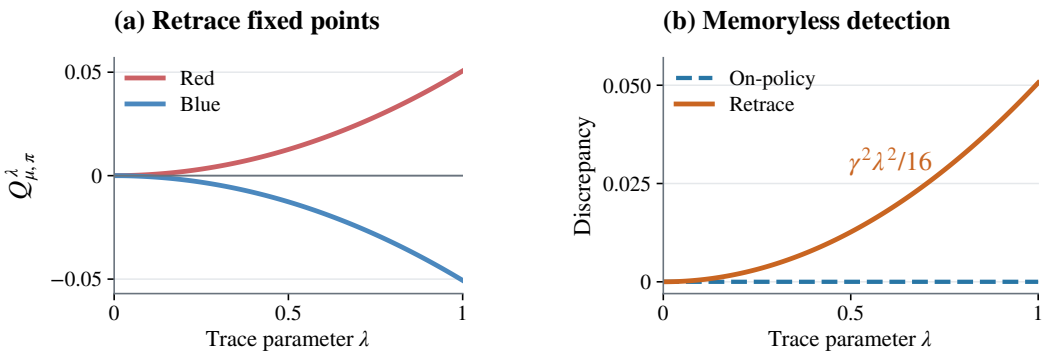

Figure 5: **Trace-length dependence of the Parity Check witness.** Exact fixed points for equation 49 at $\gamma = 0.9$. (a) Red and blue values for either action; all other observation values are zero. (b) Retrace discrepancy relative to $\lambda = 0$, alongside the zero on-policy discrepancy for the same target policy.

At white, the signed overlap is $h = 3/4 - 1/4 = 1/2$. Solving backwards from the junction, as in the preceding proof, gives for each action a

$$Q^{\lambda}_{\mu,\pi}(\text{white}, a) = Q^{\lambda}_{\mu,\pi}(\text{pink}, a) = Q^{\lambda}_{\mu,\pi}(\text{cyan}, a) = 0,$$

and

$$Q^{\lambda}_{\mu,\pi}(\text{red}, a) = \frac{\gamma^2}{2}\left[\lambda\frac{\lambda}{2} - \frac{3\lambda}{4}\frac{\lambda}{2}\right] = \frac{\gamma^2\lambda^2}{16}, \tag{50}$$
$$Q^{\lambda}_{\mu,\pi}(\text{blue}, a) = \frac{\gamma^2}{2}\left[\frac{3\lambda}{4}\frac{\lambda}{2} - \lambda\frac{\lambda}{2}\right] = -\frac{\gamma^2\lambda^2}{16}.$$

Taking the supremum norm of the difference at two trace parameters gives equation 8. Figure 5 illustrates these values and the resulting discrepancy.

The target-policy slice in Figure 1b. With behaviour and junction probabilities fixed as in equation 49, write $p_+ = \pi(\text{UP} \mid \text{pink})$ and $p_- = \pi(\text{UP} \mid \text{cyan})$. The general formula reduces to

$$D^{0,1}_{\mu,\pi} = \frac{\gamma^2}{4}\left|\left|p_- - \frac{1}{2}\right| - \left|p_+ - \frac{1}{2}\right|\right|.$$

Its zero set consists of the two diagonals $p_+ = p_-$ and $p_+ + p_- = 1$; the worked example is $(p_+, p_-) = (1/2, 3/4)$.

B.2 Dependence on a fixed memory representation

Figure 6 compares three hand-specified memories. This supplements the memoryless detection example and illustrates the symmetry-breaking role of memory also observed by Allen et al. (2024, Figure 4).

For each memory representation, lift the policies in equation 49 by ignoring the memory bit: $\mu(a \mid \omega, m) = \mu(a \mid \omega)$ and $\pi(a \mid \omega, m) = \pi(a \mid \omega)$. First consider a memory that stores $m = U$ upon the first observation and preserves it thereafter. At the junction, V remains uniform conditional on U, so $Q^{\lambda}_{\mu,\pi}((j, U), a) = 0$. At the second step, both colors are available through (v_V, U), giving

$$Q^{\lambda}_{\mu,\pi}((v_V, U), a) = \frac{\gamma\lambda}{2}UV, \qquad Q^{\lambda}_{\mu,\pi}((u_U, U), a) = 0.$$

At the first step, the weighted junction reward cancels the subtraction of the second-step value, and the remaining bootstrap averages to zero. The same formulas hold on policy, since both policies choose UP at the junction with probability $3/4$. Consequently, for this representation,

$$D^{\lambda_1,\lambda_2}_{\mu,\pi} = \Lambda^{\lambda_1,\lambda_2}_{\pi} = \frac{\gamma}{2}|\lambda_1 - \lambda_2|.$$

(a) Fixed memory updates

	u_U	v_V	j
No memory	–	–	–
First color	U	U	U
Running parity	U	UV	UV

$\gamma = 0.9, \quad (\lambda_1, \lambda_2) = (0, 1)$

(b) Discrepancy

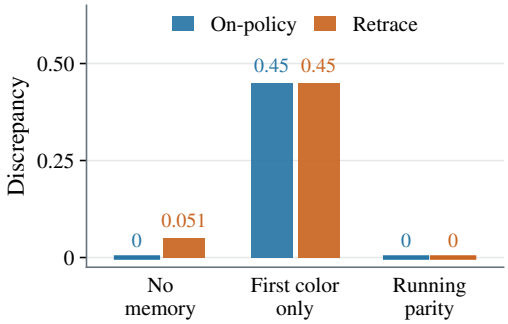

Figure 6: **Parity Check with fixed memory representations.** (a) The memory contents after each observation, with a dash denoting constant memory. (b) Exact discrepancies for the policies in equation 49, lifted to ignore the memory bit, at $\gamma = 0.9$ and $(\lambda_1, \lambda_2) = (0, 1)$. First-color memory exposes an on-policy signal that is hidden by the memoryless representation, while sufficient running-parity memory makes both discrepancies zero. Each bar describes a separate fixed representation; these are not learning trajectories.

For a running-parity memory, initialize $m = U$, update it to $m = UV$ upon the second observation, and preserve it at the junction. The fixed points are then independent of λ:

$$Q^{\lambda}_{\mu,\pi}((j, m), \text{UP}) = m, \qquad Q^{\lambda}_{\mu,\pi}((j, m), \text{DOWN}) = -m,$$
$$Q^{\lambda}_{\mu,\pi}((v_V, m), a) = \frac{\gamma m}{2}, \qquad Q^{\lambda}_{\mu,\pi}((u_U, U), a) = 0.$$

These values also hold on policy, so both discrepancies vanish. At $\gamma = 0.9$ and $(\lambda_1, \lambda_2) = (0, 1)$, the memoryless, first-color, and running-parity representations therefore have Retrace discrepancies 0.050625, 0.45, and 0, respectively; the corresponding on-policy discrepancies are 0, 0.45, and 0.

C Additional Detection Experiments

C.1 T-maze model, policies, and measurement

Latent model and observation channels. The T-maze has two equiprobable initial cue states, $2L$ corridor states (position and goal direction), two junction states, and one terminal state. East/west move along the corridor, clipping at its ends; UP/DOWN are no-ops except at the junction, where they terminate with reward 4 for the correct branch and -0.1 otherwise. Other rewards and terminal values are zero. We use $L = 5$ and an $L = 1$ control without corridor-position aliasing.

Channels in equation 9 use latent-state indices as labels: the perfect channel is the identity; corridor and junction aliasing each merge the corresponding states into one label; combined aliasing does both. Cue and terminal labels remain distinct. For stochastic mixtures we use $z = (s, \omega)$, with transition $T_{\mathcal{Z}}((s', \omega') \mid (s, \omega), a) = T(s' \mid s, a)\Phi_\eta(\omega' \mid s')$ and initial law $\zeta(s)\Phi_\eta(\omega \mid s)$, as in Appendix A.1.

Policies and fixed-point computation. Policy rows depend only on the coarse class (cue, corridor, junction), keeping physical-state action laws fixed across channel mixtures. Table 1 lists the six behaviour/target pairs; each behaviour covers its target. For each behaviour and channel, we fix the discounted-occupancy posterior (Appendix A.3) across targets and trace parameters, then solve equation 23 in double precision on all behaviour-supported nonterminal coordinates. We use $\gamma = 0.9$, $\alpha = 0$, $(\lambda_1, \lambda_2) = (0, 1)$, and $\eta \in \{0, 0.01, \ldots, 1\}$. The sweep follows Allen et al. (2024, Figure 3), replacing its $\gamma \approx 1$ and policy-weighted maximum norm with our discount and RMS score.

Table 1: Policy pairs in the complete T-maze observation-channel sweep. The target always moves east before the junction. Forward behaviours do likewise; uniform behaviour assigns probability $1/4$ to every action.

Behaviour	Behaviour UP probability at junction	Target UP probability
Forward	$2/3$	$2/3$
Forward	$1/2$	$1/2$
Forward	$1/2$	$2/3$
Forward	$1/3$	$2/3$
Uniform	$1/4$	$2/3$
Uniform	$1/4$	1

A common RMS score. Let $g_0(s)$ denote the coarse observation under combined full aliasing. Define five scored coarse observation–action pairs:

$$\mathcal{I}_0 = \{(\mathrm{cue}_+, \mathrm{E}), (\mathrm{cue}_-, \mathrm{E}), (\mathrm{corridor}, \mathrm{E}), \\ (\mathrm{junction}, \mathrm{UP}), (\mathrm{junction}, \mathrm{DOWN})\}.$$

Give each pair mass $1/5$ and distribute that mass over its latent states using the discounted occupancy d_{ref} of the forward policy with balanced junction choices:

$$\begin{aligned} w(s,a) &= \frac{\mathbf{1}\{(g_0(s),a) \in \mathcal{I}_0\}}{5} \frac{d_{\mathrm{ref}}(s)}{\sum_{y: g_0(y)=g_0(s)} d_{\mathrm{ref}}(y)}, \\ w_\eta(\omega,a) &= \sum_s \Phi_\eta(\omega \mid s) w(s,a), \\ D_{\mathrm{RMS}}(\eta) &= \left[\sum_{\omega,a:\, w_\eta(\omega,a)>0} w_\eta(\omega,a) \left(Q^1_{\mu,\pi,\eta}(\omega,a) - Q^0_{\mu,\pi,\eta}(\omega,a) \right)^2 \right]^{1/2}. \end{aligned} \tag{51}$$

The weights are policy-pair independent and sum to one for every η; at full combined aliasing, the score is uniform RMS over $\mathcal{I}_0$. Additional exploratory actions enter the fixed-point equations but not the score, so a zero score concerns only the scored coordinates.

C.2 Finite-data critic estimation

We fit neural critics from replay in the five-cell maze with combined aliasing at $\eta \in \{0, 0.5, 1\}$ and the forward behaviour/target pair with junction-UP probabilities $(1/2, 2/3)$.

At 28,672 transitions, the estimated discrepancies (mean $\pm$ standard deviation) are $(1.5 \pm 1.0) \times 10^{-4}$, 0.171 ± 0.012, and 0.158 ± 0.010 for $\eta = 0$, 0.5, and 1, respectively, compared with exact references 0, 0.173, and 0.162. Both aliased conditions remain well above the Markov control across all reported budgets and seeds.

Datasets and controlled budgets. For each seed (0–4), datasets of 64, 256, 1,024, and 4,096 seven-transition episodes are nested prefixes. Goal, action, and emission streams are independent; physical episodes and emission uniforms are paired across channels. The learner sees only observations, actions, rewards, termination flags, and policy ratios. All 60 fits use matched initial parameters within each seed, 4,000 initial updates, and 8,000 settling updates, with fixed observation encoding and policies.

Critic update and sampling convention. Each trace parameter uses two independent MLP heads with two 64-unit ReLU layers and four action outputs. Both fit detached Retrace targets; their minimum supplies estimates and bootstraps. Adam learning rates are 10^{-3} initially and 10^{-5} during settling, with exponential-moving-average target rate 0.005 and optimizer/target states carried over. Each update samples 128 complete episodes uniformly with replacement and uses all suffix returns through termination. The trace cap is one and entropy temperature zero.

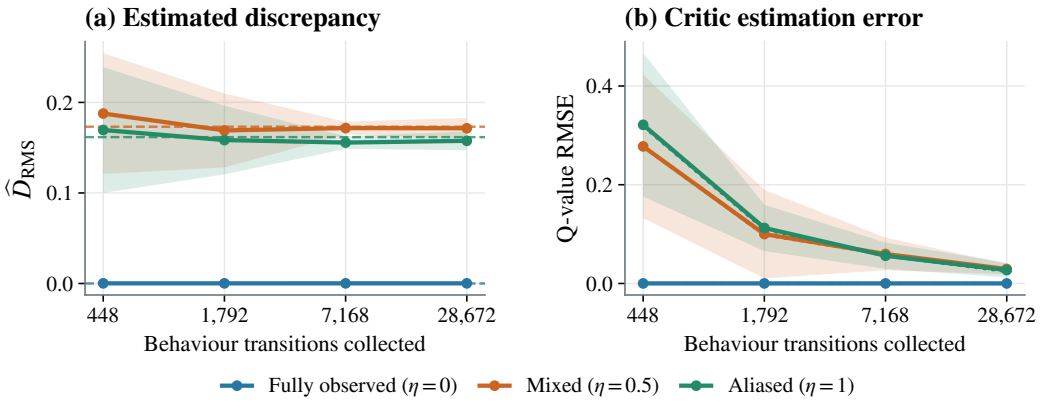

Figure 7: **Finite-data detection from off-policy replay.** (a) Estimated Retrace discrepancy; dashed lines are the corresponding exact off-policy fixed-point references. (b) Critic RMSE over both trace parameters; dotted curves give the error of the exact empirical replay fixed points on the same datasets. Curves show means over five independent seeds; bands are one sample standard deviation, truncated at zero. All quantities use original reward units.

Regression weights are proportional to γ^t and average one across the seven positions, matching the exact reference's discounted-occupancy posterior. There are no actor or discrepancy-minimization updates.

Rewards are scaled by 0.1 for fitting and predictions rescaled for evaluation; all sampled regression residuals remained in the loss's quadratic region, where its value is half the squared residual. The audit below motivated the common settling schedule; no oracle targets or per-run oracle stopping are used.

Optimization audit. An independent NumPy audit solves the empirical replay fixed point Q_{emp} using one-hot observation–action vectors x_t and forward traces initialized at $e_{-1} = 0$:

$$e_t = \gamma c_t^{\lambda} e_{t-1} + \gamma^t x_t, \qquad \sum_{\text{episodes},t} e_t \delta_t^{\pi}(q) = 0.$$

Production reverse-scan targets agree to a conditional residual below 6×10^{-8} in fitting-reward units. Comparing $\widehat{Q}$ with Q_{emp} isolates fitting error from finite-data error.

At the largest dataset, settling reduces mean value RMSEs from 0.093 to 0.029 ($\eta = 0.5$) and from 0.105 to 0.027 ($\eta = 1$), close to empirical fixed-point errors of 0.027 and 0.024. Remaining distances to Q_{emp} are 0.009 and 0.010. Figure 8 includes matched continuation controls varying only the learning rate.

Evaluation and reproducibility. The estimated discrepancy replaces the exact values in equation 51 with the online twin-min estimates. The critic error is

$$\mathrm{RMSE}_Q = \left[\frac{1}{2} \sum_{\lambda \in \{0,1\}} \sum_{\omega,a:\, w_\eta(\omega,a)>0} w_\eta(\omega,a) \left(\widehat{Q}^{\lambda}_{\mu,\pi,\eta}(\omega,a) - Q^{\lambda}_{\mu,\pi,\eta}(\omega,a) \right)^2 \right]^{1/2}.$$

Every dataset covers all scored coordinates without seed rejection or resampling. References use the same off-policy operator and sampling law.

C.3 Complete policy-pair sweeps and the one-cell control

Figures 9 and 10 extend Figure 2 to all six policy pairs and three aliasing patterns. Fully observed controls are zero to numerical precision. In the five-cell maze, corridor-position aliasing can yield a signal despite balanced junction choices. In the one-cell maze, balanced on-policy scores are

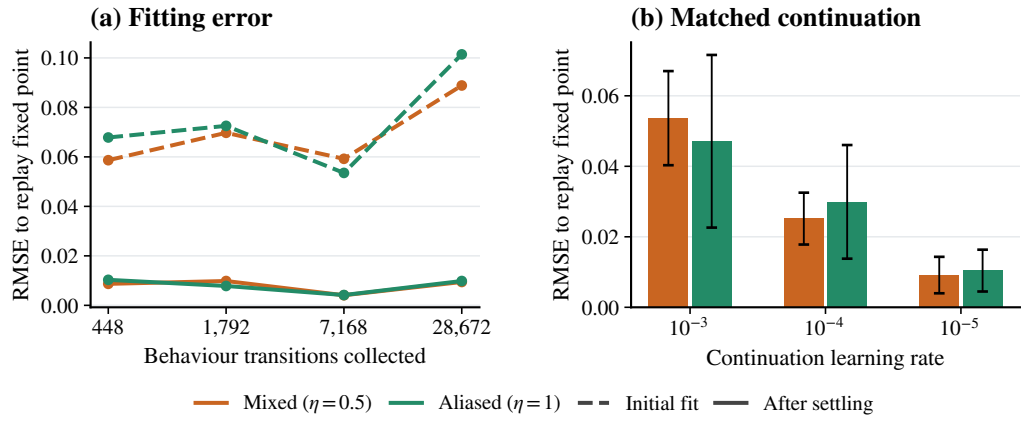

Figure 8: **Separating fitting error from finite-data error.** (a) Mean neural-critic RMSE relative to the exact empirical replay fixed point across five seeds, before and after the common settling phase. (b) Matched 8,000-update continuations from the original fits at 4,096 episodes, varying only the continuation learning rate. Bars show means and error bars one sample standard deviation for seeds 0, 2, and 4. Both panels use original reward units.

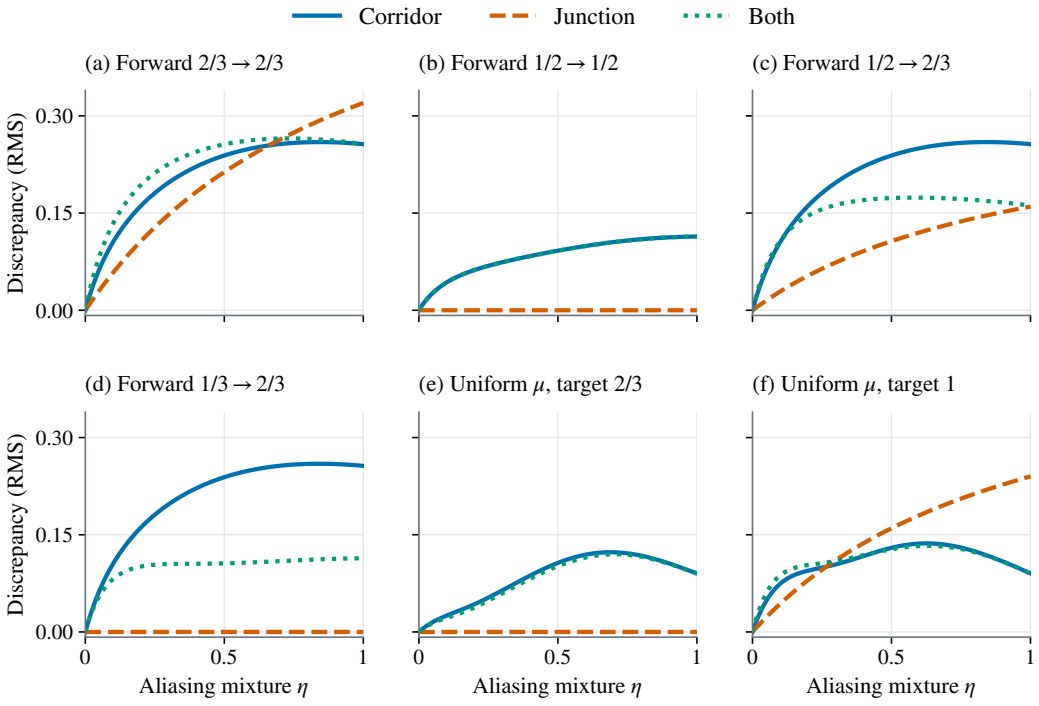

Figure 9: **Complete five-cell T-maze observation-channel sweep.** All six policy pairs in Table 1, using a shared vertical scale. Forward behaviours differ only at the junction; uniform behaviour additionally changes corridor visitation and trace continuation. Colours identify aliasing patterns, with the same convention as Figure 2.

zero throughout, but balanced behaviour detects the biased target. With a $2/3$ target and combined aliasing, reversed and uniform behaviours give positive interior scores but zero at full aliasing. These cancellations illustrate why detection is generic, not universal (Section 3.3).

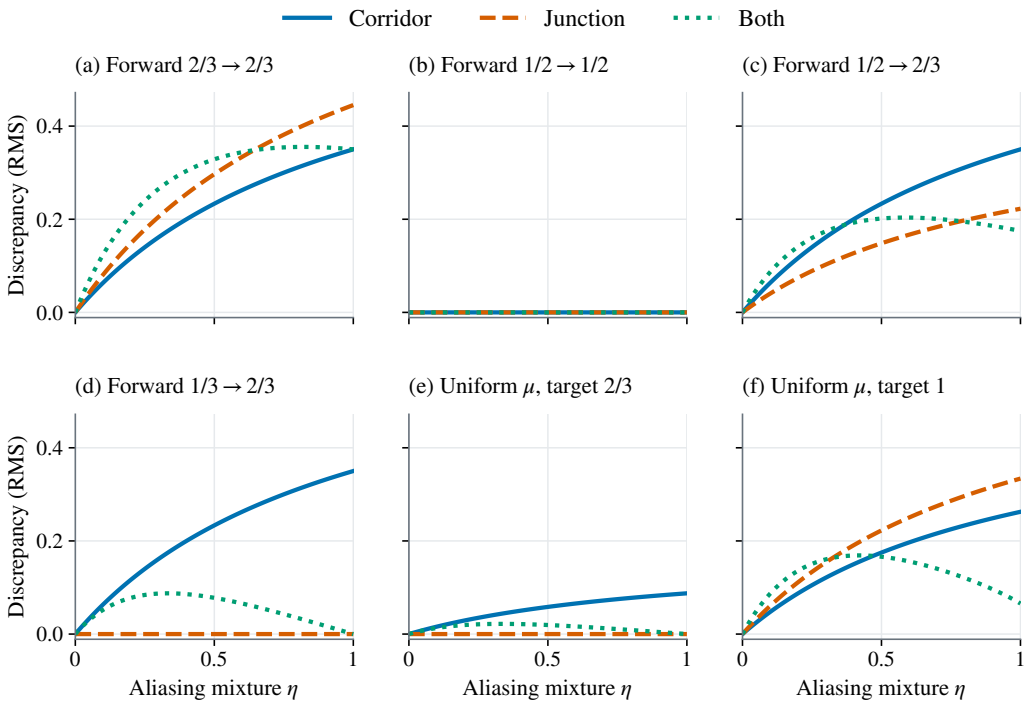

Figure 10: **One-cell T-maze control.** The same policy pairs, rewards, discount, and trace parameters as in Figure 9, with corridor-position aliasing removed. Balanced on-policy evaluation has zero discrepancy throughout the sweep. The other panels show how policy mismatch and the aliasing pattern affect detection on the scored coordinates.

C.4 Additional on-policy observation-channel controls

A further one-cell control has a nonmonotone on-policy discrepancy. The fixed policy takes east at the goal-UP cue with probability $637/9657$ (otherwise UP), always east at the goal-DOWN cue, east with probability $1/5$ in the corridor (otherwise west), and UP with probability $2/3$ at the junction. Along combined aliasing, its RMS score is zero at both endpoints but positive inside. Twelve one-at-a-time perturbations—cue-east by $\pm 0.01, \pm 0.02$, corridor-east or junction-UP by $\pm 0.02, \pm 0.04$—also have interior maxima, with positive fully aliased scores. We use 401 equally spaced mixtures and the weights in equation 51.

D Battleship Evaluation

D.1 Bayes-density reference

The reference player maintains the complete hit/miss record and knows the board dimensions and ship lengths. For each length, it enumerates horizontal and vertical placements, rejects those containing an observed miss, and assigns each remaining placement weight 12^h, where h is the number of observed hits it covers. Summing these weights over placements containing each cell gives a density score. The player fires at the highest-scoring untried cell, breaking ties by row-major index. Evaluation is greedy, with no exploration.

This baseline has explicit history and model access, but never reads hidden ship locations. Its placement-wise density approximation is a heuristic, not a proof of Bayes-optimality. We compute its expected return exactly over all 956 legal placement configurations. Weights match the environment's sequential generator: for each ship, choose orientation uniformly, then a legal nonoverlapping anchor

uniformly within that orientation. Configurations are therefore not equally likely. The expected return is 12.72 (about 13.28 shots); the reference line has no Monte Carlo estimation uncertainty.

D.2 On-policy baseline results

We use the reference implementation of Allen et al. (2024) for PPO and PPO+λ. Hyperparameter selection and the settings used in the ten-seed comparison are described in Appendix F.2.

Figure 11 shows every logged evaluation point, with no additional smoothing. The peak mean returns are 12.16 ± 0.62 for PPO and 13.48 ± 0.24 for PPO+λ; their last checkpoint means are 12.03 ± 0.39 and 13.28 ± 0.17.

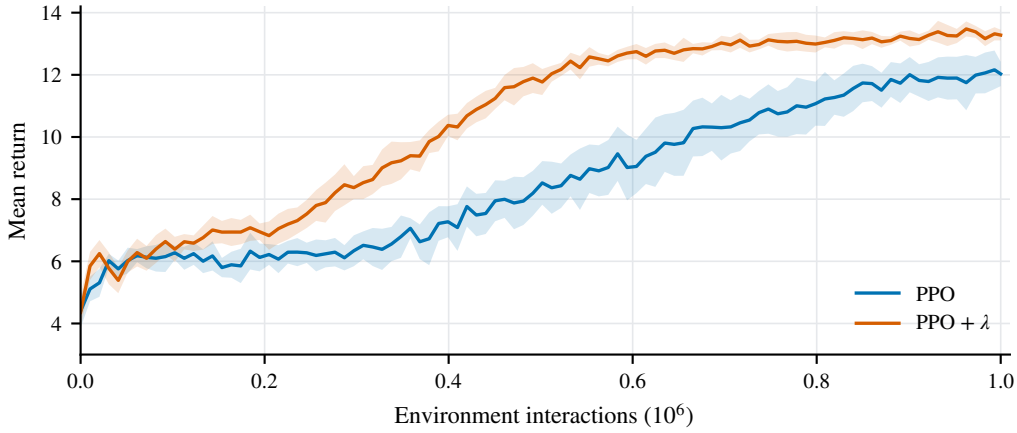

Figure 11: **Retuned on-policy Battleship baselines.** All 99 logged mean-return points for each method, with pointwise 95% CIs over ten seeds. The reference logger groups evaluations into approximately 10,000-interaction bins; no further smoothing is applied.

D.3 Evaluation statistics and interaction accounting

The agent configurations are given in Appendix F.2.

Each method is evaluated over ten training seeds. The AC+Retrace seeds are split across two jobs of five distinct seeds with matching configurations; we pool their individual trajectories at each checkpoint before computing means, confidence intervals, and whole-seed bootstrap intervals. AC returns are greedy evaluations over 50 episodes per seed every 10,000 training steps; PPO returns use the reference logger's greedy evaluations. AC curves, peak statistics, and first-passage calculations use checkpoints through 500,000 total training interactions; the on-policy reference budget is 1,000,000 interactions. We average seed-level returns and form pointwise two-sided Student-t 95% CIs using the sample standard deviation across seeds. Reported peaks select the largest observed mean; the associated intervals are pointwise, not adjusted for checkpoint selection. The memoryless run uses its available checkpoints within the AC budget.

AC's logged step count excludes random replay prefill. We add 11,904 and 12,800 transitions for the recurrent baseline and discrepancy agent, respectively, and 14,080 for the memoryless control. Counts are per seed and include all training data; evaluation and probe rollouts are excluded. For the efficiency curve, first passage uses the first recorded mean strictly above the threshold, without interpolation. We independently resample whole seed trajectories for each method in 10,000 bootstrap replicates. Unreached thresholds remain right-censored at the observation horizon; their lower and upper ratio bounds enter the percentile interval rather than being discarded. We plot empirically reached thresholds with finite interval bounds.

E MINESWEEPER EVALUATION

E.1 TASK AND OBSERVATION

We additionally evaluate memory learning on partially observed Minesweeper with a 6×6 board and six uniformly placed mines. Each action queries one cell. The observation is an eight-dimensional one-hot encoding of the latest safe cell's adjacent-mine count, with a separate reset symbol; the previous action is supplied to locate that observation on the board. The agent receives no remembered board or action mask. Repeated queries are allowed, and there is no automatic expansion of zero-clue cells or first-click protection.

With $S = 30$ safe cells, revealing a new safe cell gives reward $1/S$, repeating a safe query gives $-0.5/(S-2)$, and querying a mine gives $-0.5 - 1/S$. An episode ends on a mine, after revealing every safe cell, or after S actions. Thus previous clues and queried locations are useful for both avoiding mines and avoiding repeated actions. The compared agents are memoryless AC, recurrent AC, AC+Retrace, PPO, and PPO+λ; their selected settings are given in Appendix F.2.

E.2 EVALUATION AND SAMPLE EFFICIENCY

Every method uses twenty distinct training seeds. Each recurrent AC condition combines jobs with ten, five, and five seeds; memoryless AC combines two ten-seed jobs. We pool individual seed trajectories, giving each seed equal weight rather than each job. Both PPO conditions use twenty seeds in a single job. All splits within a method share their training settings and evaluation grid.

The AC agents train for $1{,}000{,}000$ online interactions per seed after $6{,}016$ random replay-prefill transitions. Their logged interaction counts already include this prefill, giving a final coordinate of $1{,}006{,}016$. Greedy evaluation uses 128 episodes initially and every $5{,}000$ online steps, yielding 201 checkpoints. The reference PPO agents request $2{,}000{,}000$ interactions; complete 512-transition updates give $1{,}999{,}872$ actual interactions. Their 110 evaluation checkpoints include the untrained policy and endpoint, with approximately $10{,}000$-interaction spacing through the first million and $100{,}000$ thereafter. Each checkpoint evaluates ten parallel environments for 150 steps and averages completed greedy episodes. Evaluation interactions are excluded from all training budgets.

Figure 12 uses the return and first-passage conventions of Figure 4: unsmoothed seed means, pointwise 95% Student-t CIs, and $10{,}000$ whole-seed bootstrap replicates for the experience ratio. Each method retains its full observed horizon; unreached bootstrap thresholds are right-censored rather than discarded. We plot thresholds reached by both mean curves with finite interval bounds. The ratio describes the selected learning-rate schedules and evaluation checkpoints, without interpolation between them.

The memoryless agent's peak mean return is -0.315 ± 0.011, compared with 0.025 ± 0.017 for recurrent AC and 0.039 ± 0.024 for AC+Retrace (pointwise 95% CIs). The recurrent agents substantially outperform the memoryless control; the two recurrent AC peak intervals overlap. AC+Retrace first exceeds mean return zero after $636\,016$ interactions, compared with $1\,898\,496$ for PPO+λ: an experience ratio of $3.0\times$ (95% bootstrap interval $[2.8, 3.4]$). Figure 13 shows the on-policy learning curves separately. At their longer horizon, PPO and PPO+λ finish at -0.088 ± 0.017 and 0.046 ± 0.022, respectively. The final PPO+λ mean is close to the observed AC+Retrace peak, while requiring more interaction to first reach the shared target of zero return.

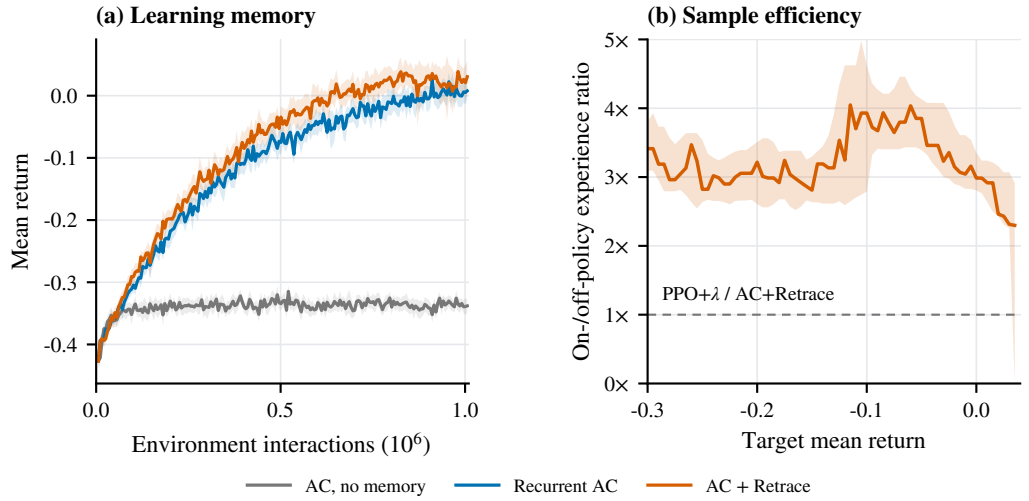

Figure 12: **Memory learning and sample efficiency in Minesweeper.** (a) Mean greedy-evaluation return for memoryless AC, recurrent AC, and AC+Retrace, with pointwise 95% CIs over twenty seeds. Curves end at 1,006,016 interactions including prefill. (b) PPO+λ interactions divided by AC+Retrace interactions to first exceed each target mean return. Shading is a 95% whole-seed bootstrap interval. Counts are per seed, and no additional smoothing is applied.

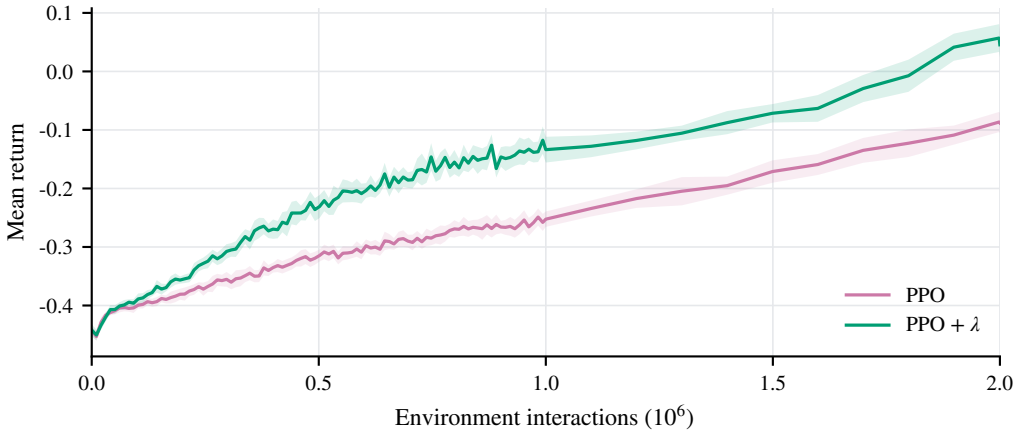

Figure 13: **Retuned on-policy Minesweeper baselines.** All 110 logged greedy-evaluation checkpoints for PPO and PPO+λ, with pointwise 95% CIs over twenty seeds and no additional smoothing. The learning rate decays linearly over the two-million-interaction training budget; complete updates give 1,999,872 actual interactions.

F Hardware and Hyperparameters

F.1 Hardware

Experiments and project development were carried out using two shared high-performance computing clusters and in-house GPU servers. We refer to the shared systems as *Cluster 1* and *Cluster 2*, omitting their names and institutional identifiers to preserve double-blind anonymity.

We used GPU compute nodes on both clusters, equipped with NVIDIA A100 GPUs with 40 GB of device memory each and AMD EPYC host processors. Table 2 summarises the node configurations according to the operators' published system specifications. Core counts refer to physical CPU cores.

These specifications describe full-node capacity, rather than the allocation required by an individual run.

Table 2: GPU compute-node configurations on the two shared clusters. GPU memory is specified per device; host memory is specified per node.

Resource	GPUs	Host CPUs	Host memory
Cluster 1	2× A100 (40 GB each)	2× EPYC 7713 (128 cores total)	512 GB
Cluster 2	4× A100 (40 GB each)	2× EPYC 7452 (64 cores total)	512 GB

Additional compute was provided by in-house servers equipped with NVIDIA H100 or H200 GPUs.

F.2 Hyperparameter selection and settings

We tuned the hyperparameters through a bayesian search in Weights&Biases[1]. Note that a search on the full set of hyperparameters did not give usable results, we therefore first tuned stability-related parameters (learning rates, entropy coefficient and target network update rate), then tuned the rest in a second sweep. Network parameters (memory type, hidden width and embedding width) and discount factor were untuned to match the reference, look-ahead tail length was chosen to guarantee residual size smaller than 0.01. We tuned to maximizing return after 100k steps in Battleship and 200k steps in Minesweeper, averaging over 3 and 5 seeds respectively.

Tables 3 and 4 report the selected configurations for the five Battleship methods in Figures 4 and 11. Tables 5 and 6 give the additional Minesweeper settings. We extract run-specific settings from the recorded training configurations and check the launch metadata; fixed implementation settings are checked against the corresponding recorded source revisions. The two five-seed Battleship AC+Retrace jobs have identical hyperparameters and contribute ten seeds to one configuration. The exact and finite-data T-maze settings are given in Appendix C.1 and its subsections.

Off-policy actor–critic. All Battleship AC variants use the latest shot outcome and previous action as network inputs, with a legal-action mask preventing repeated shots. All three variants use Adam, fixed learning rates, and the Battleship-specific embedding with hidden width 512 and one-hidden-layer actor and critic heads of width 512. The memoryless embedding has an additional feedforward layer in place of recurrence; the recurrent variants use a 512-unit GRU. Actor gradients into the shared memory are disabled in both recurrent configurations. The task and auxiliary critics use twin branches. The auxiliary heads are instantiated only for AC+Retrace, where each head uses the critic learning rate and MSE regression. Their trace parameters are $(0.05, 0.85)$, fixed in the runner; the separate successor-feature trace settings in the configuration are inactive. Actor entropy regularization has fixed coefficient 0.1 and is excluded from the task critic's bootstrap target. Replay sampling is uniform (priority exponent zero); there is no expert prefill or entropy autotuning. Burn-in starts from zero, and a 50-step look-ahead tail supplies Retrace targets. Table 3 lists the separately selected learning rates and replay sequence lengths for the three variants. The configured maximum is 100 rounds of $10{,}000$ online interactions, but all reported Battleship AC results use the common $500{,}000$-interaction window including random prefill.

On-policy hyperparameter retuning. The original Battleship configuration of Allen et al. (2024) uses a 10×10 board rather than our 5×5 board with ships $(3, 2)$, so we repeated the hyperparameter search for this setting. Directly transferring the original PPO configuration left return below six in preliminary runs, approximately matching memoryless performance. That result reflects a configuration mismatch, rather than an inability of recurrent PPO to solve the task. This pilot motivates retuning and is separate from the ten-seed curves in Figure 11. Table 4 reports the selected settings.

On-policy reference agents. Both reference agents use a 512-unit GRU with previous-action input, Adam with $\epsilon = 10^{-5}$, and linear learning-rate decay from the initial values in Table 4. The effective discount is $\gamma = 1$: the finite-horizon Battleship environment overrides the trainer's nominal discount default. For PPO+λ, the two value heads use distinct GAE trace parameters. The policy uses only the

[1] https://wandb.ai

Table 3: Selected off-policy Battleship hyperparameters. A dash denotes an inactive component. Counts are per training seed; the look-ahead tail is included in replay sequence allocation and random-prefill accounting.

Parameter	Memoryless AC	Recurrent AC	AC+Retrace
Memory	Identity	GRU	GRU
GRU hidden width	—	512	512
Embedding and head hidden width	512	512	512
Actor learning rate	10^{-4}	10^{-4}	10^{-4}
Task-critic learning rate	10^{-4}	10^{-4}	10^{-4}
Auxiliary-critic learning rate	—	—	10^{-4}
Feature-extractor learning rate	10^{-4}	10^{-5}	10^{-5}
Discount γ	0.99	0.99	0.99
Actor entropy coefficient	0.1	0.1	0.1
Target-network update rate τ	0.005	0.005	0.005
Replay capacity	100,000	100,000	100,000
Batch size (sequences)	128	128	128
Burn-in length B	10	15	10
Learning sequence length L	50	28	40
Look-ahead tail length	50	50	50
Discrepancy weight β	—	—	0.01

first head's advantage (advantage-mixture weight one). The reference discrepancy weight w mixes the squared discrepancy and value-regression losses as $w\mathcal{L}_D + (1-w)\mathcal{L}_V$, before multiplication by the value-loss coefficient 0.5. Note that the reference features a learning rate decay. The training steps are therefore important here, as longer training time would automatically inflate our sample efficiency metric. Training steps were chose such that PPO+λ reaches a similar performance as our reference. Training with a matched maximum amount of training steps leads to significantly worse final performance of 7.624 ± 0.653 (95% CI) for PPO and 10.141 ± 0.450 for PPO+λ.

Table 4: Selected on-policy Battleship hyperparameters and fixed reference-implementation settings. Learning rates are initial values; the second trace and advantage-mixture weight are inactive for PPO.

Parameter	PPO	PPO+λ
GRU hidden width	512	512
Previous-action input	Yes	Yes
Value heads	1	2
Initial learning rate	5×10^{-4}	2.5×10^{-4}
Learning-rate schedule	Linear decay	Linear decay
Adam ϵ	10^{-5}	10^{-5}
Effective discount γ	1	1
Entropy coefficient	0.01	0.05
First GAE trace parameter	0.7	0.1
Second GAE trace parameter	—	0.95
First-head advantage-mixture weight	—	1
Discrepancy mixture weight w	0	0.5
Value-loss coefficient	0.5	0.5
Policy/value clipping threshold	0.2	0.2
Maximum gradient norm	0.5	0.5
Parallel training environments	4	4
Rollout length per environment	128	128
Transitions per update	512	512
Optimization epochs per update	4	4
Minibatches per epoch	4	4
Training interaction budget	1,000,000	1,000,000

Minesweeper off-policy actor–critic. All Minesweeper agents receive the partial clue observation and previous action described in Appendix E. Recurrent agents use a 128-unit GRU; AC embeddings and single-hidden-layer heads also have width 128, with critic layer normalization. The memoryless

AC control uses identity memory with the same embedding and head widths. All AC variants stop actor gradients into the feature extractor, use uniform replay and reward-only task-critic targets, and start replay burn-in from zero. AC+Retrace uses MSE auxiliary-critic regression with $(\lambda_1, \lambda_2) = (0, 0.95)$ and $\beta = 0.1$. Table 5 lists the selected settings.

Table 5: Selected off-policy Minesweeper hyperparameters. A dash denotes an inactive component. All learning rates are constant; counts are per training seed.

Parameter	Memoryless AC	Recurrent AC	AC+Retrace
Memory	Identity	GRU	GRU
GRU hidden width	—	128	128
Embedding and head hidden width	128	128	128
Actor learning rate	10^{-4}	10^{-4}	10^{-4}
Task-critic learning rate	10^{-4}	10^{-4}	10^{-4}
Auxiliary-critic learning rate	—	—	10^{-4}
Feature-extractor learning rate	10^{-5}	10^{-5}	10^{-5}
Discount γ	0.99	0.99	0.99
Actor entropy coefficient	0.01	0.01	0.01
Target-network update rate τ	0.005	0.005	0.005
Replay capacity	100,000	100,000	100,000
Batch size (sequences)	64	64	64
Burn-in length B	30	30	30
Learning sequence length L	32	32	32
Look-ahead tail length	32	32	32
Auxiliary trace pair (λ_1, λ_2)	—	—	$(0, 0.95)$
Discrepancy weight β	—	—	0.1

Minesweeper on-policy reference agents. The reference PPO settings were selected from eight configurations per method, evaluated over three tuning seeds and one million interactions. Selection uses mean return across tuning seeds and the final five greedy evaluation checkpoints. The reported runs use twenty different training seeds and linear learning-rate decay over the selected two-million-step horizon. Both use four parallel environments, 128-step rollouts, four optimization epochs, and four minibatches per epoch, with discount 0.99, clipping threshold 0.2, value-loss coefficient 0.5, and gradient-norm cap 0.5. PPO+λ uses only the first value head's advantage and the same reference loss-mixture convention described above. These are comparisons of selected configurations and training horizons. Table 6 lists the selected settings.

Again, note that the reference features a learning rate decay. Training steps were chosen such that PPO+λ reaches a similar performance as our reference. Training with a matched maximum amount of training steps leads to significantly worse final performance of -0.151 ± 0.027 (95% CI) for PPO and -0.0580 ± 0.0388 for PPO+λ.

Table 6: Selected on-policy Minesweeper hyperparameters and fixed reference-implementation settings. Learning rates are initial values; the second trace and advantage-mixture weight are inactive for PPO.

Parameter	PPO	PPO+λ
GRU hidden width	128	128
Previous-action input	Yes	Yes
Value heads	1	2
Initial learning rate	10^{-3}	10^{-3}
Learning-rate schedule	Linear decay	Linear decay
Discount γ	0.99	0.99
Entropy coefficient	0.05	0.05
First GAE trace parameter	0.7	0.1
Second GAE trace parameter	—	0.95
First-head advantage-mixture weight	—	1
Discrepancy mixture weight w	0	0.5
Value-loss coefficient	0.5	0.5
Policy/value clipping threshold	0.2	0.2
Maximum gradient norm	0.5	0.5
Parallel training environments	4	4
Rollout length per environment	128	128
Transitions per update	512	512
Optimization epochs per update	4	4
Minibatches per epoch	4	4
Training interaction budget	2,000,000	2,000,000

G TRAINING OBJECTIVES FOR RECURRENT REPLAY LEARNING

The five losses in equation 10 use replay averages $\langle\cdot\rangle$ after burn-in. Superscript $\min$ denotes the minimum over twin branches $k \in \{1, 2\}$; action sums use the legal action set.

$$\mathcal{L}_{\text{actor}} = \left\langle \sum_a \pi_\varphi(a \mid m_t) \left[\alpha \log \pi_\varphi(a \mid m_t) - Q_\phi^{\min}(m_t, a)\right] \right\rangle, \tag{52}$$

$$\mathcal{L}_Q = \frac{1}{2} \left\langle \sum_{k=1}^{2} \left(Q_{\phi_k}(m_t, a_t) - y_t^Q\right)^2 \right\rangle, \tag{53}$$

$$\mathcal{L}_{\lambda_1} = \frac{1}{2} \left\langle \sum_{k=1}^{2} \left(q_{\psi_{1,k}}^{\lambda_1}(m_t, a_t) - y_t^{\lambda_1}\right)^2 \right\rangle, \tag{54}$$

$$\mathcal{L}_{\lambda_2} = \frac{1}{2} \left\langle \sum_{k=1}^{2} \left(q_{\psi_{2,k}}^{\lambda_2}(m_t, a_t) - y_t^{\lambda_2}\right)^2 \right\rangle, \tag{55}$$

$$\mathcal{L}_D = \left\langle \left(q_{\psi_1}^{\lambda_1,\min}(m_t, a_t) - q_{\psi_2}^{\lambda_2,\min}(m_t, a_t)\right)^2 \right\rangle. \tag{56}$$

Targets are detached: $y_t^Q = r_{t+1} + \gamma(1 - d_t)\bar{V}_{t+1}^Q$, where d_t marks episode termination and $\bar{V}^Q$ is the expected twin-min value under the target policy and target networks. $y_t^{\lambda_j}$ is the truncated Retrace target from equation 5, using auxiliary critic j and stored behaviour probabilities. Both targets are reward-only; auxiliary regression excludes the look-ahead tail. For recurrent agents, the actor loss updates only the actor head, critic losses update their heads and memory, and the discrepancy updates memory with auxiliary-head parameters frozen.