Memory-State Critic for Asymmetric Actor-Critic with Application to Vision-Based Pursuit-Evasion

Arthur Louette[1] **Alejandro Sánchez Roncero**[2] **Gaspard Lambrechts**[3] **Pascal Leroy**[1,4]
Julien Hansen[1] **Petter Ögren**[2] **Damien Ernst**[1]

[1]University of Liège, Belgium [2]KTH Royal Institute of Technology, Sweden
[3]McGill University & Mila Québec AI Institute, Canada [4]Belerion, Belgium
arthur.louette@uliege.be

Abstract

In partially observable Markov decision processes, the optimal policy generally depends on the history of observations and past actions. Asymmetric actor-critic methods have become popular to learn such policies when additional information, such as the true state of the environment, is available during training. The critic, which is not needed at execution, is given access to the state. A critic conditioned on the state alone is generally ill-defined and yields biased policy gradients. Conditioning on the state and the history, the history-state critic restores both. In this paper, we show that conditioning the critic on the state and the policy's own memory, i.e., the internal representation of the history through which the policy selects its actions, is already well-defined and gives unbiased policy gradients, removing the need for a second recurrent approximator of the history. We call it the *memory-state critic*. It follows that a critic based on the policy's memory need not backpropagate its loss into that memory, even though the memory is a lossy encoding of the history. We evaluate the memory-state critic in a vision-based pursuit-evasion environment between two quadrotors across two arena types. The pursuer is the learning agent, and the evader is sampled per episode from a fixed pool of heuristic behaviours. The results show that the memory-state critic outperforms the history-state critic and converges faster. In addition to being unbiased compared to the state-only critic, it maintains a slight edge in the wall arena, where the actor's history carries information that the privileged state alone does not.

1 Introduction

In many real-world decision problems, such as robotic control problems, an agent must take actions based on noisy or incomplete observations of the underlying state of its environment. Such problems are modelled as partially observable Markov decision processes (POMDPs) [Åström, 1965]. Reinforcement learning (RL) in POMDPs addresses such decision problems through trial and error to maximise a reward [Sutton et al., 1998]. A consequence of this partial observability is that the optimal control strategy at any given time usually depends on the entire history of observations and past actions. In POMDPs, the history-dependent strategy, or policy, is usually implemented with a recurrent neural network (RNN) that recurrently updates an internal state, called a hidden state, based on successive observations and actions [Ni et al., 2022]. The hidden state is trained by backpropagating the RL loss through time, not by predicting states or observations. It therefore encodes what the policy needs from the history to act. Throughout the paper, we employ state for the environment state of the POMDP and *memory* for the policy's recurrent hidden state.

19th European Workshop on Reinforcement Learning (EWRL 2026).

Many successful RL algorithms for POMDPs adopt the asymmetric learning paradigm. It takes foundation in the idea is that, in simulation, the true state of the environment is available at training time. Pinto et al. [2018] introduced the asymmetric actor-critic algorithm in which the critic accesses the state only during training, while the policy accesses the observation. This algorithm has been shown to work better in some environments and is widely used in sim-to-real robot learning [Kaufmann et al., 2023, Degrave et al., 2022, Vasco et al., 2024].

Baisero and Amato [2022] proved that a critic that conditions only on the state is generally ill-defined in a POMDP. Indeed, the state alone is not a sufficient statistic of the future execution of the POMDP, i.e., two different histories reaching the same state can induce different action distributions, since the policy conditions on the history. Instead, the history-state critic, which takes as input both the history and the state, is well-defined because the history and state together are a Markovian variable of the future execution of the POMDP. In addition, the history-state critic is an unbiased estimate of the history critic, and it can be used in the policy gradient. In practice, the critic cannot take the full history either. It has to encode it with an encoder such as a recurrent network of its own, which is trained by the value loss rather than the policy gradient. Having a critic that takes the history as input has the disadvantage that it needs a second encoder of the history in comparison to the state-only critic of Pinto et al. [2018]. It doubles the recurrent computation at training time and induces a second network.

In this work, we introduce the *memory-state critic*, which conditions on the state and the policy's own memory. The critic takes as input the policy's memory, but the critic loss is not backpropagated through the recurrent encoder of the history. We prove that the memory-state critic gives the same on-policy value function and the same unbiased policy gradient as the history-state critic. Indeed, the memory is a sufficient statistic of the history for future actions. In the symmetric setting where the critic does not see the state, reusing the actor's hidden state alone would not give an unbiased gradient. Indeed, in that setting, the memory of the agent is not a Markovian variable of the future execution of the POMDP, since it is not necessarily a sufficient statistic for future states of the environment. However, the combination of the state and the actor's memory keeps the policy gradient unbiased.

We evaluate the memory-state critic on a vision-based pursuit-evasion environment between two quadrotors. The pursuer is the learning agent. It observes the world through an onboard camera and must catch an evader that regularly leaves the camera frame during maneuvers. The evader is sampled per episode from a fixed pool of four heuristic behaviours, and this information is part of the critic's privileged state, not of the policy's observation. The evader has a stationary policy, and the problem reduces to a single-agent POMDP. We evaluate on two arenas that differ in their degree of partial observability: an open arena with no obstacles and the same arena with a wall to increase the degree of partial observability. We compare the memory-state critic to three baselines: a state-only critic, a history-state critic, and a reactive critic that conditions on the state and the current observation and action. The memory-state critic matches the history-state critic in the final return and converges faster, as expected from the removal of the second recurrent encoder. It also marginally outperforms the state-only critic in the wall arena, where the state aliases the history. We release the environment and our implementation[1], built on IsaacLab [Mittal et al., 2023] and skrl [Serrano-Muñoz et al., 2023].

The remainder of the paper is organised as follows. Section 2 reviews POMDPs and the asymmetric policy gradient. Section 3 introduces the memory-state critic and proves its two properties. Section 4 reports the experiments, which are analysed in Section 5. The paper concludes with Section 6 discussing the related work and conclusions of our work 7.

2 Background

2.1 POMDPs and history-dependent policies

A POMDP is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{O}, T, O, R, P, \gamma)$. In a state $s_t \in \mathcal{S}$, the agent collects observations $o_t \in \mathcal{O}$ via $O(o_t \mid s_t)$ at time t and selects actions a_t based on the observable history $h_t = (o_0, a_0, \ldots, o_t)$. We denote the set of histories by $\mathcal{H} = \bigcup_{t=0}^{\infty} \mathcal{H}_t$ with $\mathcal{H}_t \subseteq \mathcal{O} \times (\mathcal{A} \times \mathcal{O})^t$. After taking action $a_t \in \mathcal{A}$, the state transitions according to $T(s_{t+1} \mid s_t, a_t)$. The immediate reward is a scalar $r_t = R(s_t, a_t)$, the initial state s_0 is drawn from $P(s_0)$. The discount factor $\gamma \in [0, 1)$ weights the importance of future rewards. A policy $\pi(a_t \mid h_t)$ maps a history to a distribution over actions, and is said to be

[1] `https://github.com/LouetteArthur/MemoryStateCritic`

optimal when it maximises the expected return $J(\pi) = \mathbb{E}^\pi[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)]$. We denote the future return from time t by $G_t = \sum_{k=0}^{\infty} \gamma^k R(s_{t+k}, a_{t+k})$.

2.2 Value functions and the asymmetric policy gradient

The canonical value function in a POMDP is the history value function $V^\pi(h_t) = \mathbb{E}^\pi[G_t \mid h_t]$, with history-action value function $Q^\pi(h_t, a_t) = \mathbb{E}^\pi[G_t \mid h_t, a_t]$. The history h_t is a sufficient statistic of the future execution under π, so $V^\pi(h_t)$ is well-defined. Baisero and Amato [2022] introduce the *history-state value function* $V^\pi(s_t, h_t) = \mathbb{E}^\pi[G_t \mid s_t, h_t]$ and the corresponding $Q^\pi(s_t, h_t, a_t) = \mathbb{E}^\pi[G_t \mid s_t, h_t, a_t]$, which additionally condition on the state. We therefore have $V^\pi(h_t) = \mathbb{E}_{s_t|h_t}[V^\pi(s_t, h_t)]$ and, likewise, $Q^\pi(h_t, a_t) = \mathbb{E}_{s_t|h_t}[Q^\pi(s_t, h_t, a_t)]$.

In contrast, the state alone is not a sufficient statistic of the process: two histories reaching the same state can induce different action distributions, since the policy conditions on the history. As a result, the time-invariant state value function $V^\pi(s)$ is generally ill-defined. Even when well-defined, the resulting policy gradient is biased [Baisero and Amato, 2022].

For a parameterised history-dependent policy π_θ, Baisero and Amato [2022] showed that the policy gradient can be written equivalently with either value function as a baseline:

$$\begin{aligned} \nabla_\theta J(\pi_\theta) &= \mathbb{E}^\pi \left[\sum_t \gamma^t \nabla_\theta \log \pi_\theta(a_t \mid h_t)\, A^\pi(h_t, a_t) \right] \\ &= \mathbb{E}^\pi \left[\sum_t \gamma^t \nabla_\theta \log \pi_\theta(a_t \mid h_t)\, A^\pi(s_t, h_t, a_t) \right], \end{aligned} \tag{1}$$

where $A^\pi(h, a) = Q^\pi(h, a) - V^\pi(h)$ and $A^\pi(s, h, a) = Q^\pi(s, h, a) - V^\pi(s, h)$. The asymmetric variant on the right uses the extra information s in the critic without changing the policy gradient.

2.3 Recurrent actor-critic architectures

In practice, the size of the history h_t grows linearly in t which becomes impractical. The standard approach is to encode it through a recurrent neural network. A function $z_t = f(h_t)$ is a *recurrent encoder* when it admits an update rule $z_t = u(z_{t-1}, a_{t-1}, o_t)$ from an initial hidden state z_0 and an arbitrary null action a_0. The encoder is deterministic given its inputs, as is standard for recurrent networks. The policy is then parameterised as

$$\pi_\theta(a_t \mid h_t) = g_\theta(a_t \mid f_\theta(h_t)) = g_\theta(a_t \mid z_t^a), \tag{2}$$

where f_θ is the recurrent encoder and g_θ is an action head producing the action distribution. We write $z_t^a = f_\theta(h_t)$ for the policy's memory at time t.

The history-state critic of Baisero and Amato [2022] is implemented analogously, with its own recurrent encoder $z_t^c = f_\psi(h_t)$ on the critic side and an MLP head producing $V_\psi(s_t, z_t^c)$. The two encoders are trained in parallel, f_θ by the policy gradient and f_ψ by the value loss, typically with TD learning. Figure 1 illustrates this design. It duplicates the recurrent computation at training time, and creates a second representation-learning problem on the critic side. The representation z^c learned by the critic may also diverge from z^a learned by the policy, since the two are trained from different losses. We show in the next section that this second encoder is not necessary.

3 The memory-state critic

3.1 Definition

Let $z_t^a = f_\theta(h_t)$ be the policy's memory. We define the *memory-state value function* and its action-value counterpart as

$$V^\pi(s_t, z_t^a) = \mathbb{E}^\pi[G_t \mid s_t, z_t^a], \qquad Q^\pi(s_t, z_t^a, a_t) = \mathbb{E}^\pi[G_t \mid s_t, z_t^a, a_t]. \tag{3}$$

The conditioning event is now the pair (s_t, z_t^a) of the state and the policy's memory, rather than the pair (s_t, h_t) of the state and the full history. We will show that, on-policy, the two value functions agree, and that $V^\pi(s, z^a)$ can be used in the asymmetric policy gradient (1) without bias.

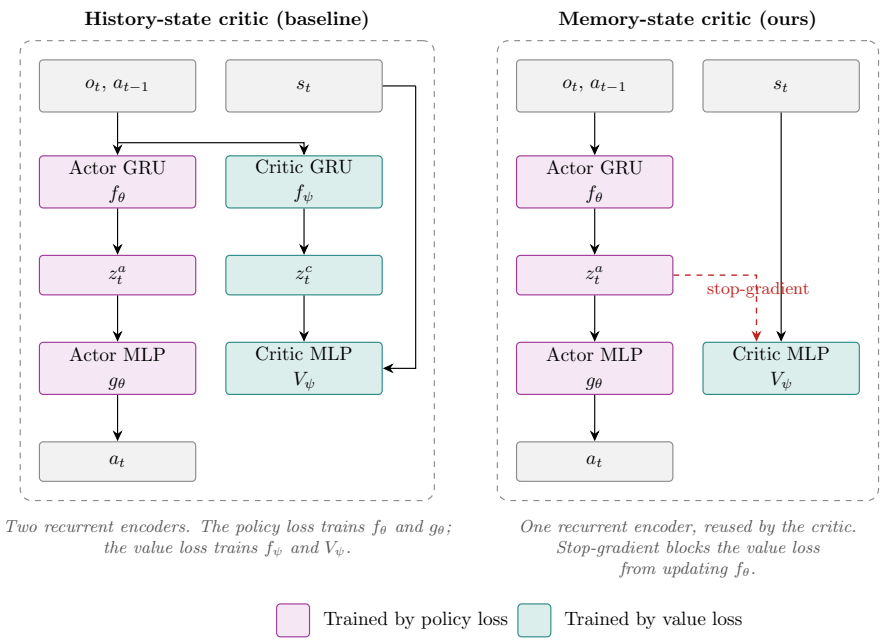

Figure 1: Asymmetric actor-critic architectures with a recurrent policy. Left: the history-state critic maintains a second recurrent encoder f_ψ on the critic side, in parallel with the policy's encoder f_θ. Right: the memory-state critic reuses the policy's hidden state z^a with a stop-gradient, removing the second recurrent module. Both critics consume the state s in their MLP head.

3.2 On-policy equality

Proposition 1 (On-policy equality)**.** *Let f_θ be a deterministic recurrent encoder, and $\pi_\theta(a \mid h) = g_\theta(a \mid f_\theta(h))$ the corresponding recurrent policy. Then, for all reachable (h_t, s_t),*

$$V^\pi(s_t, z_t^a) = V^\pi(s_t, h_t) \quad \textit{and} \quad Q^\pi(s_t, z_t^a, a_t) = Q^\pi(s_t, h_t, a_t) \quad \textit{with } z_t^a = f_\theta(h_t). \tag{4}$$

Proof. We hereafter show that conditioning on (s_t, z_t^a) and conditioning on (s_t, h_t) induce the same on-policy distribution over future trajectories $(s_{t+k}, a_{t+k})_{k\geq 0}$. The policy's action satisfies $a_t \sim g_\theta(\cdot \mid z_t^a)$ by the factorisation (2), so it depends on h_t only through z_t^a. The transition $s_{t+1} \sim T(\cdot \mid s_t, a_t)$ depends on h_t only through a_t, hence only through z_t^a. T he next observation $o_{t+1} \sim O(\cdot \mid s_{t+1})$ depends on h_t only through s_{t+1}, hence only through (s_t, z_t^a). The deterministic encoder update $z_{t+1}^a = u(z_t^a, a_t, o_{t+1})$ depends on h_t only through (z_t^a, a_t, o_{t+1}), hence only through (s_t, z_t^a). By induction on the trajectory length, the on-policy distributions of $(s_{t+k}, a_{t+k})_{k\geq 0}$ conditioned on (s_t, z_t^a) and on (s_t, h_t) coincide. The future-return distributions therefore agree, and the two value functions are equal. The same induction, started from a fixed a_t instead of $a_t \sim g_\theta(\cdot \mid z_t^a)$, gives the action-value statement. □

3.3 Asymmetric policy gradient with the memory-state value function

Proposition 2 (Asymmetric policy gradient)**.** *Under the same assumptions as Proposition 1, the asymmetric policy gradient can be written as*

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}^\pi\left[\sum_t \gamma^t \nabla_\theta \log \pi_\theta(a_t \mid h_t)\, A^\pi(s_t, z_t^a, a_t)\right], \tag{5}$$

where $A^\pi(s_t, z_t^a, a_t) = Q^\pi(s_t, z_t^a, a_t) - V^\pi(s_t, z_t^a)$.

Proof. We show that both advantages coincide, so the substitution leaves the gradient unchanged. Step 1 first records that $V^\pi(s, z^a)$ meets the baseline requirement of Baisero and Amato [2022].

Step 1: $V^\pi(s, z^a)$ averages back to the history value. Baisero and Amato [2022] prove that $V^\pi(h_t) = \mathbb{E}_{s_t|h_t}[V^\pi(s_t, h_t)]$. Substituting Proposition 1,

$$\mathbb{E}_{s_t|h_t}[V^\pi(s_t, z_t^a)] = \mathbb{E}_{s_t|h_t}[V^\pi(s_t, h_t)] = V^\pi(h_t). \tag{6}$$

The memory-state value function is therefore an unbiased estimate of the history value function.

Step 2: the two advantages coincide. Proposition 1, applied also to the action-value function, makes the two advantages coincide pointwise on every reachable (s_t, h_t, a_t):

$$A^\pi(s_t, z_t^a, a_t) = Q^\pi(s_t, z_t^a, a_t) - V^\pi(s_t, z_t^a) = Q^\pi(s_t, h_t, a_t) - V^\pi(s_t, h_t) = A^\pi(s_t, h_t, a_t). \tag{7}$$

The two expectations are therefore taken of the same random variable, and (5) follows. □

The state is essential. Both propositions rely on having the state s_t in the conditioning. Without it, the policy's memory z_t^a alone is not a sufficient statistic for the future return. Two histories that map to the same z^a can correspond to different states, and the state in turn drives the future transitions through $T(\cdot \mid s_t, a_t)$. Lambrechts et al. [2025] gives a finite-time convergence analysis of asymmetric critics with linear function approximators, in which the state appears as exactly the quantity that eliminates aliasing in the agent state. Proposition 1 is the analogous statement, the state closes the aliasing left by the memory.

3.4 Approximation in practice

The two value functions $V^\pi(s_t, h_t)$ and $V^\pi(s_t, z_t^a)$ are analytical objects. In practice, both are approximated by neural networks trained with TD learning. The memory-state critic uses the same architecture for the value head but reads its recurrent input from the policy's encoder:

$$V_\psi(s_t, z_t^a) \quad \text{with } z_t^a = f_\theta(h_t). \tag{8}$$

The critic shares f_θ in the forward pass. The critic loss is not backpropagated through f_θ. The encoder f_θ is updated only by the policy gradient. A stop-gradient placed on z_t^a in the input to the critic implements this. Without the stop-gradient, the value loss would flow into the policy's encoder and shape z^a to minimise value error alongside the policy loss, which is known to destabilise on-policy training [Cobbe et al., 2021]. The resulting architecture is shown on the right of Figure 1. By comparison, the history-state critic of Baisero and Amato [2022] is approximated by $V_\psi(s_t, z_t^c)$ with $z_t^c = f_\psi(h_t)$, where f_ψ is a critic-side recurrent encoder with parameters ψ disjoint from θ. Both f_θ and f_ψ are trained, the first by the policy gradient and the second by the value loss.

4 Experiments

4.1 Environment

We evaluate the memory-state critic on a vision-based pursuit-evasion environment between two Crazyflie quadrotors. The pursuer is the only learning agent. It must get within a 20 cm radius of the evader and within 10 seconds to neutralise it. Otherwise, the pursuer gets penalised. The arena is a $5 \times 4 \times 2$ metre box. We use two geometries that differ in their degree of partial observability. The *open arena* has no obstacles, so the evader stays visible whenever it is inside the camera frame. The *wall arena* contains an opaque vertical wall with a gap, which can hide the evader from the pursuer even when the camera points at it. The wall arena is the environment where the theory predicts the largest gap between critics, because occlusion creates histories that the privileged state alone cannot disambiguate.

The pursuer observes a 64×64 two-channel image from a camera, with a depth channel and a binary opponent-segmentation mask, together with its previous action. A visualisation of the pursuer input is presented in Figure 2. The policy is therefore conditioned on the history. The critic additionally sees a 64-dimensional privileged state s_t with the state of both drones, relative-pose features, propeller speeds, and an identifier of the heuristic drawn from the evader pool at the start of the episode. The pursuer acts through a four-dimensional body-rate and collective-thrust command at 25 Hz, tracked by an inner PD loop at 500 Hz. Dynamics are modelled at the propeller level. Implementation details are in Appendix A.

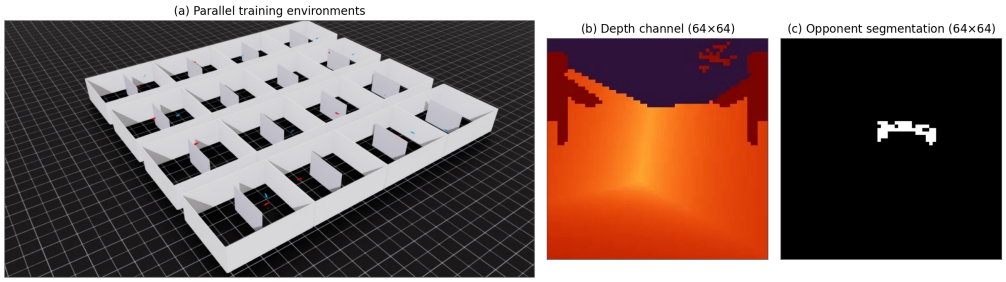

Figure 2: Vision-based pursuit-evasion environment. Left: IsaacLab simulates many arenas in parallel. Each contains a pursuer (red), an evader (blue), and an optional wall with a gap that can hide the evader. Middle: a representative depth channel from the pursuer's camera. Right: the binary opponent segmentation mask. Both channels are 64×64.

The evader is sampled per episode from a fixed pool of four heuristic behaviours: a stationary hover, a circular trajectory, a lemniscate trajectory, and an artificial-potential-field reactive policy[Ma'Arif et al., 2021]. Each heuristic acts as a function of the state, and its identifier at episode start is part of s_t. From the pursuer's point of view, the evader is therefore part of the environment, and the setting reduces to a single-agent POMDP that satisfies the assumptions of Proposition 1. The construction does not directly apply when the opponent is a learned recurrent policy that conditions on its own history and does not take into account that the other might still be learning. We leave it to future work.

Let d_t be the inter-agent distance, replaced by a wall-aware distance $\tilde{d}_t$ in the wall arena. Let δ_t^+ be the capture flag, b_t^P and w_t^P the pursuer out-of-bounds and wall-collision flags, v_t a visibility flag that equals one only when the evader is visible, and ρ_t the angular offset of the evader from the camera axis. The pursuer's reward is

$$r_t^P = \underbrace{\kappa_a \left[\Phi(\tilde{d}_t) - \Phi(\tilde{d}_{t-1})\right]}_{\text{approach}} + \underbrace{\kappa_p \, v_t \, e^{-\kappa_\rho \rho_t}}_{\text{visibility}} - \underbrace{1/T + R\,\delta_t^+}_{\text{time and capture}} - \underbrace{R\,(b_t^P + w_t^P)}_{\text{crash}} - \kappa_{\text{br}} \| W_{\text{br}} \omega_t \|, \quad (9)$$

with approach potential $\Phi(d) = \exp(-d/d_0)$. The visibility term rewards keeping the evader near the camera axis, but only when it is visible, so the agent cannot earn it by pointing at an occluded evader. Coefficient values are in Appendix A.

4.2 Architecture and training

We integrate the memory-state critic into proximal policy optimization[Schulman et al., 2017], extended for recurrent actors, building on skrl[Serrano-Muñoz et al., 2023]. The actor is a convolutional encoder followed by a GRU and an action head. It consumes the onboard camera observation and the previous action, and outputs a continuous action distribution. The critic is a multi-layer perceptron that takes the privileged state s_t concatenated with the recurrent encoded history chosen by the critic variant. For the memory-state critic, the recurrent encoded history is the actor's hidden state z_t^a. For the history-state critic, it is a critic-side encoder f_ψ with independent weights that generates the hidden state z_t^c. Both encoders share the same architecture. For the wall arena, we use a slightly larger discount factor $\gamma = 0.999$ to provide a longer credit-assignment horizon around the wall gap. All other hyperparameters are unchanged. Hyperparameters are in Appendix A.

4.3 Critic variants and protocol

We compare four critics, listed in Table 1. They differ only in the input of the critic. The actor architecture and all other hyperparameters are identical across variants. The observation-state critic tests whether a recurrent encoder on the critic side is necessary at all or if observation and previous action are enough. The state-only and history-state critics are the baselines from prior work. We train each critic variant with proximal policy optimization for 5.12×10^7 environment steps under the same hyperparameters (Appendix A). We run 20 seeds per variant for $V_\psi(s)$, $V_\psi(s, o, a)$, $V_\psi(s, z^c)$, and $V_\psi(s, z^a)$ in the open and wall arenas. We report aggregate performance using the interquartile

mean (IQM) across seeds Agarwal et al. [2022]. Bands are 95% bootstrap confidence intervals over seeds.

Table 1: Critic variants compared in the experiments.

Critic	Symbol	Description
Observation-state	$V_\psi(s, o, a)$	State, current observation, previous action.
State-only	$V_\psi(s)$	State only. Biased with a policy conditioned on the history.
History-state	$V_\psi(s, z^c)$	State and a critic-side recurrent encoding of the history.
Memory-state (ours)	$V_\psi(s, z^a)$	State and the policy's memory with stop-gradient.

5 Results

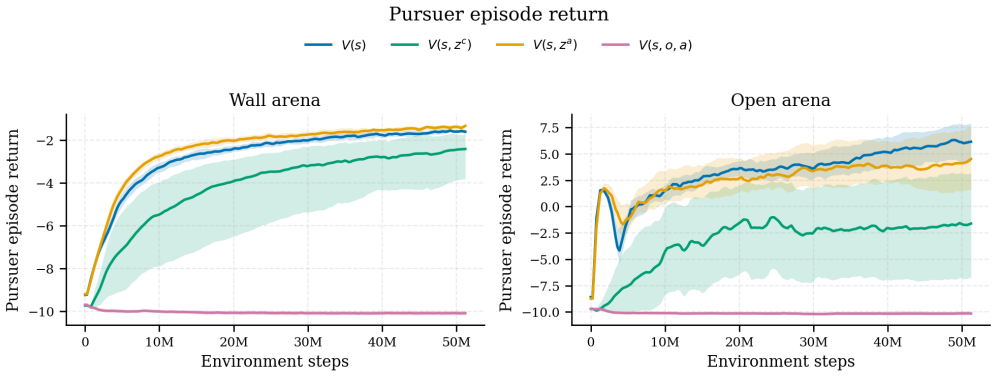

Figure 3: Pursuer episode return during training. Lines are interquartile means over 20 seeds [Agarwal et al., 2022]. The shaded intervals are 95% bootstrap confidence intervals. The memory-state critic matches or exceeds the history-state critic in both arenas and converges faster. It exceeds the state-only critic in the wall arena and matches it in the open arena.

The memory-state critic outperforms the history-state critic and converges faster. Figure 3 reports the episodic return during training. In both arenas, the memory-state critic reaches at least the final return of the history-state critic in fewer steps. This is the outcome demonstrated by Proposition 1. In the open arena, the history-state critic plateaus lower and has way more variance across seeds. We attribute this to its second optimisation problem, since its encoder must learn from the value loss alone a representation predictive of future return. Therefore, it should be predictive of the actor's next action without observing the actor's own representation, which is what the memory-state critic reuses for free. While these theoretical and empirical findings are promising, they are currently limited in terms of environment diversity. Because the construction holds for any asymmetric POMDP with a deterministic recurrent actor, comparing it on standard POMDP benchmarks is a natural next step.

The bias of the state-only critic matters only when the state aliases the history. Propositions 1 and 2 guarantee that the critics are unbiased, but not that they beat the biased state-only critic in every environment. The size of the bias scales with how much the privileged state aliases the history [Lambrechts et al., 2025]. In the open arena, the evader is visible whenever it is in frame, and the observation model is deterministic, so the state is close to Markov for the future return, and the state-only critic is nearly unbiased. In addition, the state-only critic has an easier architecture to train. Its return is therefore comparable to the memory-state critic's, consistent with it solving a smaller learning problem. In the wall arena, the evader is repeatedly occluded, so two histories that reach the same state can disagree on where the evader was last seen. The state-only critic cannot represent this difference, in contrast to the history-state and memory-state critic. This could explain why the memory-state critic marginally performs better in the wall arena and the history-state critic is closer to the other in episode return

The reactive critic did not learn in our environment. The reactive critic $V_\psi(s, o, a)$ replaces the recurrent memory with the current observation and previous action, and fails to learn a useful policy

in either arena (Figure 3). The current observation and previous action carry far less information about the return than the history. This confirms that the memory term, and not merely the state, is what recovers the history-state performance.

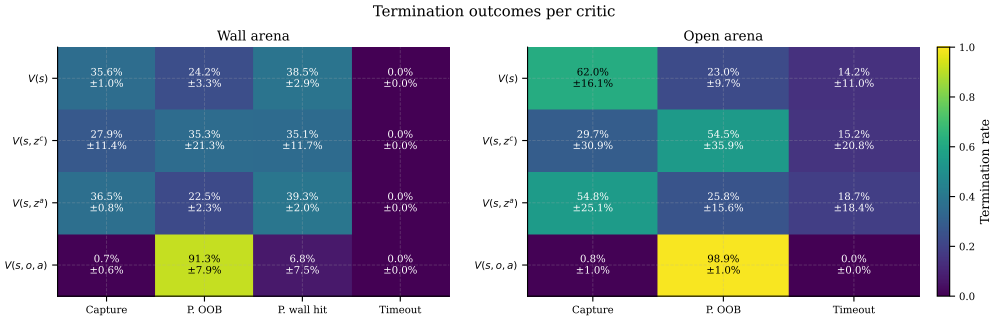

Figure 4: Terminal-condition breakdown per critic and arena. Each cell shows the fraction of evaluation episodes ending in the given outcome as the mean over 20 seeds, $\pm$ one standard deviation. The wall arena adds the pursuer wall collision outcome.

Termination analysis. Figure 4 decomposes how episodes end. Among the critics that learns to capture the evaders, the history-state critic has the highest pursuer out-of-bounds (OOB) rate in the open arena, consistent with its lower return, as the actor overshoots the evader rather than tracking it. The reactive critic $V_\psi(s, o, a)$ crashes (equivalent to OOB) in almost every episode, but it never learns to capture. The memory-state critic achieves the best capture rate in the wall arena, where occlusion makes the privileged state alias the history. In the wall arena, pursuer wall collisions account for roughly 35 to 40% of episodes for all learning critics, indicating that wall avoidance is a hard subproblem independent of the critic's history representation.

6 Related work

Asymmetric actor-critic. Pinto et al. [2018] introduced the asymmetric actor-critic algorithm for image-based robot learning, with a reactive policy and a state-only critic. Baisero and Amato [2022] prove that with a history-dependent policy, the time-invariant state value function is generally ill-defined, and that even when well-defined the resulting policy gradient is biased. They propose the history-state critic and show that it yields an unbiased policy gradient. Lyu et al. [2022] extend the analysis to multi-agent reinforcement learning. Lambrechts et al. [2025] give a finite-time convergence analysis with linear function approximators. Ebi et al. [2025] generalise the construction to partial privileged information beyond the full state. Our work provides a deep RL implementation of the history-state critic in which the policy's recurrent encoder is reused on the critic side, removing the second encoder.

Recurrent policies for POMDPs. Recurrent policies are a standard way to handle partial observability, and are competitive with transformer-based variants in RL Ni et al. [2022], Hausknecht and Stone [2015]. The history-dependent policy in our work follows this design. Our contribution is on the critic side.

Vision-based drone control. Kaufmann et al. [2023] train a vision-based drone racing policy in simulation with an asymmetric actor-critic, with the policy relying on state estimation from off-board sensors at training time. Geles et al. [2024] remove the state-estimation step and train an end-to-end vision policy with a reactive actor. The questions addressed in this work do not arise in their setting because their policy is non-recurrent. Drone racing also exhibits little aliasing, because the next gate rarely leaves the camera frame, in contrast to pursuit-evasion. Model-based vision RL has shown strong results in single-agent racing [Romero et al., 2026, Verraest et al., 2025] and is complementary to our work.

Quadrotor pursuit-evasion. Roncero et al. [2025] and Hansen et al. [2026] train quadrotor pursuit-evasion policies under full state observation in a multi-agent setting. Bajcsy et al. [2024]

consider vision-based pursuit-evasion with a ground robot evader. To the best of our knowledge, our environment is the first open-source vision-based quadrotor pursuit-evasion with propeller-level dynamics.

7 Conclusion

We introduced the memory-state critic, a single-encoder asymmetric critic for recurrent actors in POMDPs. It uses the actor's recurrent hidden state, which avoids learning a second statistic of the history for the value function. We proved that it gives the same on-policy value function and unbiased policy gradient as the history-state critic of Baisero and Amato [2022], with one recurrent encoder instead of two. In a vision-based pursuit-evasion environment between two quadrotors, the memory-state critic outperforms the history-state critic in final performance and converges faster. It also marginally outperforms the state-only critic, especially when partial observability is important. Sim-to-real transfer of vision-based drone policies is an active area [Yu et al., 2025], and physical deployment is future work.

Acknowledgments

Gaspard Lambrechts is a *Marie Skłodowska-Curie Actions* (MSCA) postdoctoral fellow at McGill University in Canada, funded by *Horizon Europe* under grant 101284668. Gaspard Lambrechts also gratefully acknowledges the financial support of *Wallonia-Brussels International* for his *World Excellence Fellowship*.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice, 2022. URL `https://arxiv.org/abs/2108.13264`.

Karl Johan Åström. Optimal control of markov processes with incomplete state information i. *Journal of mathematical analysis and applications*, 10:174–205, 1965.

Andrea Baisero and Christopher Amato. Unbiased asymmetric reinforcement learning under partial observability. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 44–52, 2022.

Andrea Bajcsy, Antonio Loquercio, Ashish Kumar, and Jitendra Malik. Learning vision-based pursuit-evasion robot policies. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9197–9204, 2024. arXiv:2308.16185.

Karl Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. Phasic policy gradient. In *International Conference on Machine Learning (ICML)*, pages 2020–2027. PMLR, 2021.

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022. doi: 10.1038/s41586-021-04301-9.

Daniel Ebi, Gaspard Lambrechts, Damien Ernst, and Klemens Böhm. Informed asymmetric actor-critic: Theoretical insights and open questions. In *European Workshop on Reinforcement Learning (EWRL)*, 2025. URL `https://openreview.net/forum?id=wNVclrpGfU`.

Ismail Geles, Leonard Bauersfeld, Angel Romero, Jiaxu Xing, and Davide Scaramuzza. Demonstrating agile flight from pixels without state estimation, 2024. URL `https://arxiv.org/abs/2406.12505`.

Julien Hansen, Arthur Louette, Pascal Leroy, and Damien Ernst. Autonomous surveillance drone: Learning pursuit-evasion interception with population-based IPPO. *Under review*, 2026.

Matthew J Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. In *AAAI fall symposia*, volume 45, page 141, 2015.

Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620: 982–987, 2023. doi: 10.1038/s41586-023-06419-4.

Gaspard Lambrechts, Damien Ernst, and Aditya Mahajan. A theoretical justification for asymmetric actor-critic algorithms. In *International Conference on Machine Learning (ICML)*, pages 32375–32405, 2025. arXiv:2501.19116.

Xueguang Lyu, Andrea Baisero, Yuchen Xiao, and Christopher Amato. A deeper understanding of state-based critics in multi-agent reinforcement learning. In *AAAI Conference on Artificial Intelligence*, pages 9396–9404, 2022.

Alfian Ma'Arif, Wahyu Rahmaniar, Marco Antonio Márquez Vera, Aninditya Anggari Nuryono, Rania Majdoubi, and Abdullah Çakan. Artificial potential field algorithm for obstacle avoidance in uav quadrotor for dynamic environment. In *2021 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, pages 184–189, 2021.

Mayank Mittal, Calvin Yu, Qinxi Yu, Jingzhou Liu, Nikita Rudin, David Hoeller, Jia Lin Yuan, Ritvik Singh, Yunrong Guo, Hammad Mazhar, Ajay Mandlekar, Buck Babich, Gavriel State, Marco Hutter, and Animesh Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023. doi: 10.1109/LRA.2023.3270034.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free rl can be a strong baseline for many pomdps, 2022. URL `https://arxiv.org/abs/2110.05038`.

Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. *RSS*, 2018.

Angel Romero, Ashwin Shenai, Ismail Geles, Elie Aljalbout, and Davide Scaramuzza. Dream to fly: Model-based reinforcement learning for vision-based drone flight, 2026. URL `https://arxiv.org/abs/2501.14377`.

Alejandro Sanchez Roncero, Yixi Cai, Olov Andersson, and Petter Ogren. Learned controllers for agile quadrotors in pursuit-evasion games, 2025. URL `https://arxiv.org/abs/2506.02849`.

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

Antonio Serrano-Muñoz, Dimitrios Chrysostomou, Simon Bøgh, and Nestor Arana-Arexolaleiba. skrl: Modular and flexible library for reinforcement learning. *Journal of Machine Learning Research*, 24(254):1–9, 2023.

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

Miguel Vasco, Takuma Seno, Kenta Kawamoto, Kaushik Subramanian, Peter R. Wurman, and Peter Stone. A super-human vision-based reinforcement learning agent for autonomous racing in Gran Turismo. *Reinforcement Learning Journal*, 4:1674–1710, 2024.

Aderik Verraest, Stavrow Bahnam, Robin Ferede, Guido de Croon, and Christophe De Wagter. SkyDreamer: Interpretable end-to-end vision-based drone racing with model-based reinforcement learning. *arXiv preprint arXiv:2510.14783*, 2025.

Hang Yu, Christophe De Wagter, and Guido C. H. E. de Croon. Depth transfer: Learning to see like a simulator for real-world drone navigation, 2025.

A Appendix

Quadrotor dynamics. We model each quadrotor as a rigid body of mass m and inertia tensor $\mathbf{J}$, with state $(p, v, R, \omega, \Omega)$ for position, velocity, attitude, body rates, and the four rotor speeds. The continuous dynamics follow Kaufmann et al. [2023]. Each rotor produces thrust $k_T\Omega_i^2$ and moment $k_M\Omega_i^2$, and the rotor speeds follow first-order dynamics $\dot{\Omega}_i = k_{\text{mot}}(\Omega_{\text{ss},i} - \Omega_i)$ towards the commanded steady-state speed $\Omega_{\text{ss},i}$.

Action space and low-level control. The policy outputs a four-dimensional action $a \in [-1, 1]^4$ at 25 Hz, three commanded body rates and a collective thrust. Body rates are scaled to $\omega_k^{\text{ref}} = \pi\, \omega_k^{\text{cmd}}$ rad/s, and the thrust to $T^{\text{ref}} = \frac{1}{2}(T^{\text{cmd}} + 1)\, mg\, \eta_{\text{T2W}}$ with $\eta_{\text{T2W}} \approx 3.15$. An inner PD controller at 500 Hz tracks these references and produces the per-rotor setpoints.

Camera. Each drone carries a forward-facing pinhole camera rendered with IsaacLab's tiled-rendering pipeline, at 64×64 resolution and a 120° horizontal field of view. We read two channels, depth and a binary opponent-segmentation mask, with a clipping range 0.02 to 20 m.

Reward and arena. The reward coefficients in (9) are $R = 10.0$, $\kappa_a = 3.0$, $d_0 = 0.5$ m, $\kappa_p = 0.02$, $\kappa_\rho = 1.0$, and $\kappa_{\text{br}} = 0.001$, with capture distance $r_c = 0.2$ m and maximum episode length $T = 250$ steps. The arena bounds are $[-2.5, 2.5] \times [-2.0, 2.0] \times [0.0, 2.0]$ m. The wall in the occluded arena is 0.05 m thick with a 1.5 m gap, and the drone-wall clearance is 0.15 m. The body-rate weighting is $W_{\text{br}} = \text{diag}(1, 1, 0.2)$, which lowers the yaw penalty to allow reorientation.

Table 2: PPO hyperparameters, shared across all critic variants.

Parameter	Value
Discount factor γ	0.99 (open) / 0.999 (wall)
GAE λ	0.95
Learning rate	3×10^{-4}
LR scheduler	KL-adaptive ($\tau_{\text{KL}} = 0.008$)
Rollout length	32 steps
Mini-batches	4
Learning epochs	8
Clip ratio ϵ	0.2
Value clip	0.2
Entropy coefficient	0.01
Value loss coefficient	2.0
Gradient norm clip	1.0
Reward scaling	0.1
Initial log σ	0.0
GRU hidden size	256
GRU layers	1
Backpropagation sequence length	16
Parallel environments	512
Policy rate	25 Hz
Simulation rate	500 Hz
Episode length	10 s